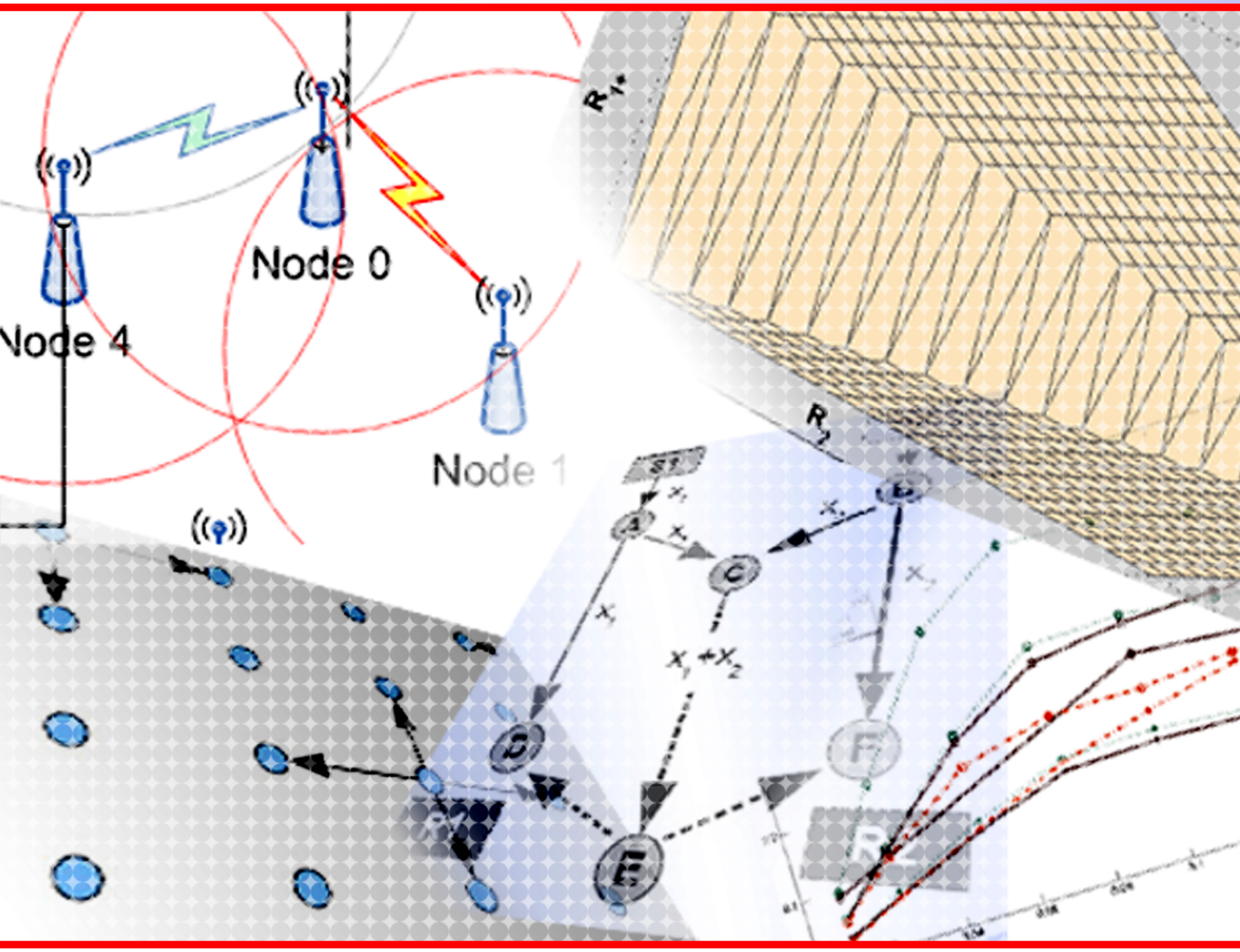
Masters Thesis

MAC/Routing Layer Interaction with Wireless Network Coding



Bayu Anggoro Jati
Nurul Huda Mahmood

Aalborg University

Masters Thesis

MAC/Routing layer interaction with Wireless Network Coding

Group No. 07gr1120

Bayu Anggoro Jati
Nurul Huda Mahmood

June 2007
AALBORG UNIVERSITY

Dedication

To our Parents...
whose selfless and delicate nurturing
made us who we are today

Aalborg University
E-Studyboard
Mobile Communications 10th semester

TITLE:

MAC/Routing layer interaction with Wireless Network Coding

THEME:
Mobile Radio Communications

PROJECT PERIOD:
1st February 2007 - 21th June 2007

PROJECT GROUP:
07gr1120

GROUP MEMBERS:
Bayu Anggoro Jati
Nurul Huda Mahmood

SUPERVISORS:
Hiroyuki Yomo
Petar Popovski

Number Of Duplicates: 5

Number Of Pages In Report: 56

Number Of Pages In Appendix: -

Total Number Of Pages: 69

Abstract

Network Coding offers a new degree of freedom for improving performance in a communication network. With this, multiple bi-directional information flows are merged and processed as a single flow instead of treating them separately. So unlike conventional systems, networks employing network coding will prefer flows to be routed such that there is an increased interaction among the flows. Routing techniques with such end in view is termed as *Coding aware routing* in the literature.

In this thesis, we propose, implement and evaluate two such coding aware routing algorithms. One is a simple algorithm called *Highway based routing*, where flows are routed through a set of pre-determined *highways*, thus maximizing interaction among the flows. The other is the more complex *adaptable routing algorithm* that tries to maximize positive interaction of a newly arrived flow with the existing flows in the network. We demonstrate the throughput gain of network coding over conventional transmission analytically and through simulations. Network coding with these proposed 'coding aware routing' algorithms result in as high as 50 % throughput gain over network coding with conventional routing techniques.

Contents

List of Figures

Acronyms

AoDV	Ad-Hoc On demand Distance Vector
BWA	Broadband Wireless Access
COPE	Coding Opportunistically
DSDV	Destination-Sequenced Distance-Vector
DSR	Dynamic Source Routing
FIFO	First In First Out
LAN	Local Area Network
IP	Internet Protocol
MAC	Medium Access Control
WiFi	Wireless Fidelity
WiMAX	Worldwide Interoperability for Microwave Access
WMN	Wireless Mesh Network
XOR	eXclusive OR
ACK	Acknowledgment
WLAN	Wireless Local Area Network

Preface

This Master's Thesis by Bayu Anggoro Jati and Nurul Huda Mahmood is submitted to Aalborg University as partial fulfillment of the requirements for obtaining Master's Degree in Mobile Communications. Work for this Master's project was done between 1st February to 7th June 2007 under the supervision of Associate Professor Hiroyuki Yomo, and Assistant Professor Petar Popovski from the Department of Communication Technology, Aalborg University, Denmark.

This project report documents the study and investigation on MAC and routing layer interaction with wireless network coding. It proposes a novel approach to routing in wireless networks aimed at maximizing network performance with wireless network coding. Results of both analytical study and implementation by simulation are presented in this thesis.

The basic concepts of network coding is introduced in the Introduction, which is then expounded in details in the following Chapter. Chapter three introduces wireless mesh networks and highlights some widely used existing routing techniques in wireless mesh network. Findings and observations of analytical and practical study on wireless network coding in a simple linear network is presented next, followed by a similar study with two-dimensional grid network architecture in Chapter five. Finally the work is summed up in the concluding chapter.

Throughout the report figures are enumerated successively in each chapter. The first time a word of special interest for the project is used, we emphasize it. References are given on the form [reference number], where 'reference number' is listed in the bibliography at the very end.

Bayu Anggoro Jati

Nurul Huda Mahmood

June 2007
Aalborg University

Acknowledgement

"Glory be to Thee (Oh Lord)! we have no knowledge saving that which Thou hast taught us; surely Thou art perfect in knowledge and wisdom" (The Holy Quran, Chapter 2: Verse 32)

Our heartfelt gratitude and sincerest acknowledgement to Associate Professor Hiroyuki Yomo and Assistant Professor Petar Popovski for their constructive supervision and relentless support in seeing this project through.

We graciously celebrate all our beloved family and friends, whose constant guidance and encouragement continues to nourish our journey through life, as it did with this project.

Chapter 1

Introduction

Almost all existing communication systems today have some, if not all of their entities connected together through a network, be it large such as the telephone network and Internet, or a small Local Area Network (LAN) of few computers. Irrespective of the size or nature, information packet flow in these networks is achieved by transmitting packets from source to destination possibly through a number of intermediate nodes along a determined route, keeping independent information streams separated from each other. This separation of independent flows greatly limit the amount of information that can flow through a network for a given source/destination pair.

Recently this conventional idea was challenged as *"folk wisdom"* in the seminal paper [2]. Instead, the authors proposed the novel idea of *Networking Coding* as a new approach to look into network information flow. With Network coding information from different flows at an intermediate node are combined and transmitted together. The receiver can then decode the packet intended for itself from this combined or encoded packet using information it holds. This multiple packet transmission in a single instance dramatically improves network performance in terms of throughput, delay and robustness among others. Initially network coding was proposed for wired network with multi-cast communication, but recent work shows that network coding has many applications in a wireless multi-hop network as well due to the inherent broadcast nature of wireless communication, and can significantly improve its performance [3–6].

1.1 Motivation

Advances in computer processing, communication systems and electronics have resulted in the booming of wireless networking technologies, bringing it at par with traditional communication services. Wireless networks can either be infrastructure based or ad-hoc (infrastructure-less). In infrastructured wireless network, a base-station, which is a wireless transceiver, acts as a local hub connecting together the wireless network devices. In Ad-hoc wireless networks on the other hand, wireless nodes directly connect to other nodes in the network that are within its range. Figure 1.1 shows an example of these two type of wireless networks.

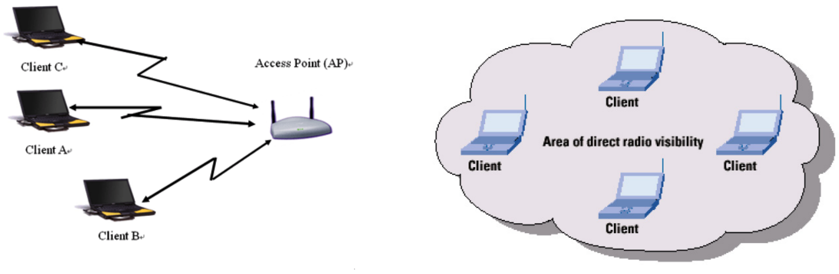


Figure 1.1: Example of Infrastructureand ad-hoc wireless network

Wireless Mesh Network (WMN) is another wireless networking technology that builds on the concept of ad-hoc network to support more advanced functionalities. Because of its flexibility & mobility, plug-and-play convenience and low cost, Wireless Mesh Networks are becoming an indispensable part of today's communication technology [7].

Because of the nature of wireless media, only one user can access a wireless channel at any given time. It is also an expensive and limited resource with each user having right to use at most a few channels only. This significantly limits the amount of information a wireless network can carry simultaneously. Moreover, its distributed nature and randomness of topology at times introduce a lot of information flow in the network resulting in heavy congestion. These shortcomings of conventional transmission mechanism result in a severe throughput limitation. It is suggested that this limitation can be addressed by employing 'Network Coding', which facilitates intermixing multiple information flows to enhance overall throughput in a wireless multi hop network [1].

The concept of network coding can be explained by the illustration iof Figure 1.2 below.

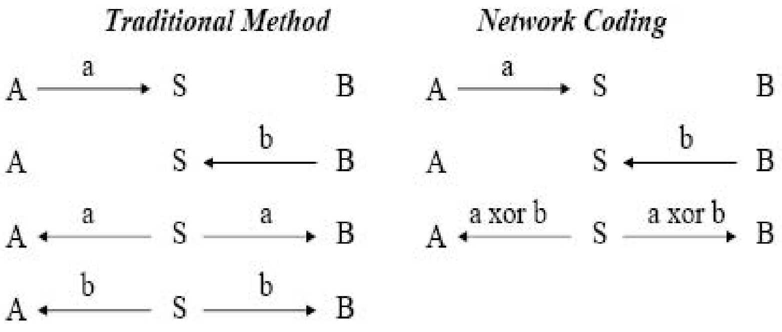


Figure 1.2: Simple network coding example [1]

Suppose both nodes A and B want to send a packet to each other. Both of them are using node S as an intermediate node to reach the destination. With traditional method, this kind of information exchange needs four transmissions whereas with network coding it can be done with only three transmissions. The difference between the traditional and network coding method is that with network coding, the packet which is sent by A and B are XOR'ed in the node S. Afterwards, the XOR'ed packet is broadcasted to both A and B. Node A can decode the packet b from Node B because it already has packet a; and similarly for Node B. As mentioned earlier, it reduces one transmission compared to the traditional one, thus increasing throughput of the network. Furthermore, due to less transmission required, battery power is saved and a higher degree of robustness is achieved [1].

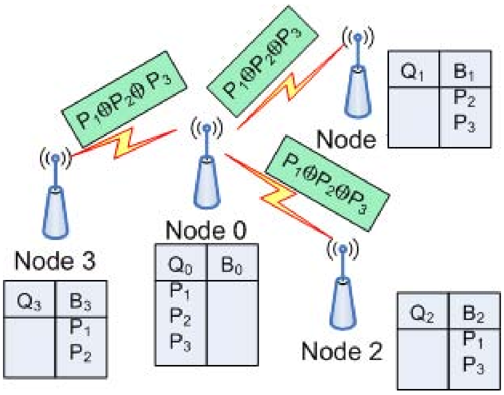


Figure 1.3: Illustration of how network coding works

The throughput gain with network coding is achieved by combining several packets intended for different destinations in a single packet that is to be broadcast. This is shown in Figure 1.3. For each intended receiving nodes to be able to successfully decode its packet, it is required that each nodes have all packets other than the packet intended for it. For example, suppose a node N_0 has packets p_1, p_2 and p_3 intended for nodes N_1, N_2 and N_3 respectively. Now it can generate an encoded packet by combining packets p_1, p_2 and p_3 and transmit it over a single time slot, if and only if each of Nodes N_i (i = 1,2,3) has packet p_j (j = 1,2,3), $j \neq i$. On the other hand, the intended receiver N_1 of packet p_1, will be able to decode the packet p_1 from the encoded packet by doing an XOR operation since it has the other two packets, p_2 and p_3. Hence the *usability* of network coding techniques relies on the fact that each node has many packets to encode and decode information from.

This means that networks employing network coding would like information flows in the network to overlap and create opportunities for coding (see Figure 1.4).

Thus current routing mechanisms and techniques, which are designed to minimize interaction between information flows, will be radically challenged and has to be rethought all new [8]. Existing conventional routing algorithms focus on avoiding congestion by transmitting packets through sparsely congested nodes as much as possible, whereas network coding can handle more congestion and would in fact prefer networks being congested to a certain degree. This calls for routing decisions to be made based with the awareness of network coding, introducing the concept of *'coding aware routing'*.

In a preliminary evaluation of coding-aware routing algorithms made in [8], it is found to offer significant gain in terms of throughput and reduced number of transmission. This

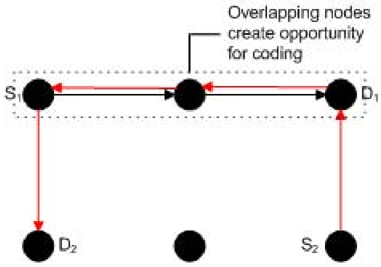


Figure 1.4: Example of overlapping Information flow

has motivated us to look deeper into this problem and propose some routing algorithms catered for network coding.

1.2 Problem Definition

In this thesis, we focus on wireless mesh and multi-hop network which consists of many nodes with wireless interface. As we can see, wireless network coding has potential to significantly improve the throughput. However, as explained above, routing algorithms in wireless multi hop networks employing network coding will have to strike a balance between supporting network coding and avoiding throughput degradation and excessive delay due to congestion. Thus, there can be a tradeoff/interaction among network coding, routing, and MAC layers operation.

The goal of this project is to analyze the above trade-off when we apply Decode and Forward (DF) [3] based wireless network coding to wireless mesh network, and propose join optimization strategy to enhance the throughput performance. Decode and Forward technique is one of the wireless network coding techniques which involve eXclusive OR (XOR) operation on data packets, as explained in previous subsection. We choose this specific network coding technique since it is easier to incorporate in the current protocols compared to others. The main objectives of this study are:

- To investigate the routing strategies which can take advantage of DF-based wireless network coding
- To investigate the interaction of the gain/cost brought by routing, MAC and the DF-based wireless network coding
- To propose join optimization method of MAC/routing operations for the DF-based wireless network coding

These will be done under different network scenarios. We assume the nodes are static i.e. there is no mobility.

1.2.1 Scenarios

Within the scope of routing and MAC layer interaction with network coding, it is important to limit the work of this project to certain scenarios to be able to make a meaningful study in the given time frame. In this project we focus on the following two scenarios:

First Scenario

The first scenario we will investigate is a linear collection of nodes, as shown in Figure 1.5 below. In this structure, any two nodes are connected through only one possible route, and so the transmission scheduling and route between nodes is fixed.

Figure 1.5: The linear network structure

The goal in this scenario is to implement DF-enabled network coding and observe throughput gain achieved due to network coding. Studies will also be done to explore how different information flows affect network coding, and therefore the throughput gain. The expected outcome of this study is observation on the interaction between information flow and throughput gain due to network coding. Details of this scenario and findings from this study is presented in Chapter 4.

Second Scenario

The second scenario under study in this project is the grid structure. Here the nodes are arranged in a $N \times M$ wireless mesh structure, with all nodes equidistant from each other. The basic setup of grid structure is shown in Figure 1.6 below.

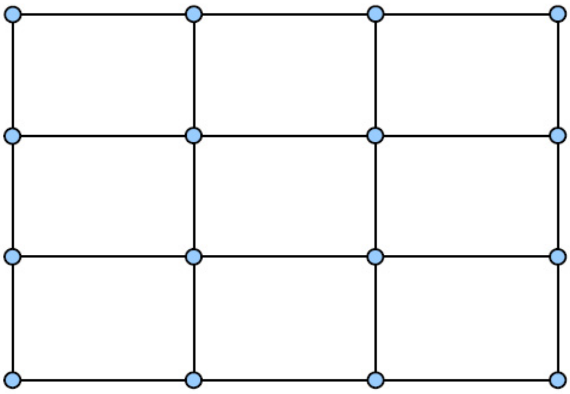


Figure 1.6: The grid network structure

Like the linear structure, we adopt a fixed centrally controlled scheduling here as well. We later extend this to the case of dynamic scheduling to do cross-layer optimization, where nodes with more transmission load are given higher chance to transmit. The routing, however, is not fixed like in the case of linear scenario, and the routes for each flow has to be discovered. This is a major challenge of this scenario, where a route has to be determined that optimizes the coding gain as well as overall network throughput. In this scenario, we will try a few different routing algorithms.

The purpose of studying this scenario is to observe the interaction between scheduling and routing layer, and how different routing algorithms and scheduling mechanisms would impact network coding and the throughput gain. The expected outcome of this study is to determine an optimized routing algorithm with network coding. This scenario is explained with further details in Chapter 5.

Working tools

Investigation for both of the above described scenarios is done both analytically and with computer simulation. The analytical work determines whether a set of given flows with a given offered load per flow can be supported in each of the above scenarios, and if supported, how? The empirical work, carried out using simulation tools in $MATLAB^{\circledR}$ and C++, is aimed at implementing the routing algorithms to support a set of given flows.

1.3 Outline of this report

The next part of this report will begin with giving in detail description about network coding technology, and its state of the art. Then results and findings of the first scenario, i.e. the linear structure, is presented in Chapter 4. Detailed description of the grid structure is presented in the next chapter followed by results, discussion and evaluation. The report is then finally concluded in the final Chapter 6.

Chapter 2

Network Coding

Network coding is a recent field in information theory that proposes a radical new outlook to information flow in a network. Though initially proposed for wire-line network and multi-cast transmission, recent studies have shown that there are promising applications to wireless network with unicast transmission.

In network coding, information from different flows towards different direction at a given node can be encoded by combining together, and transmitted as a single broadcast packet. The receiving nodes then decode the packet and extract required information using knowledge about past flows that they have [1].

We may present the famous example of butterfly network [2] shown in Figure 2.1 to illustrate the basic concept of network coding and expected benefits therein.

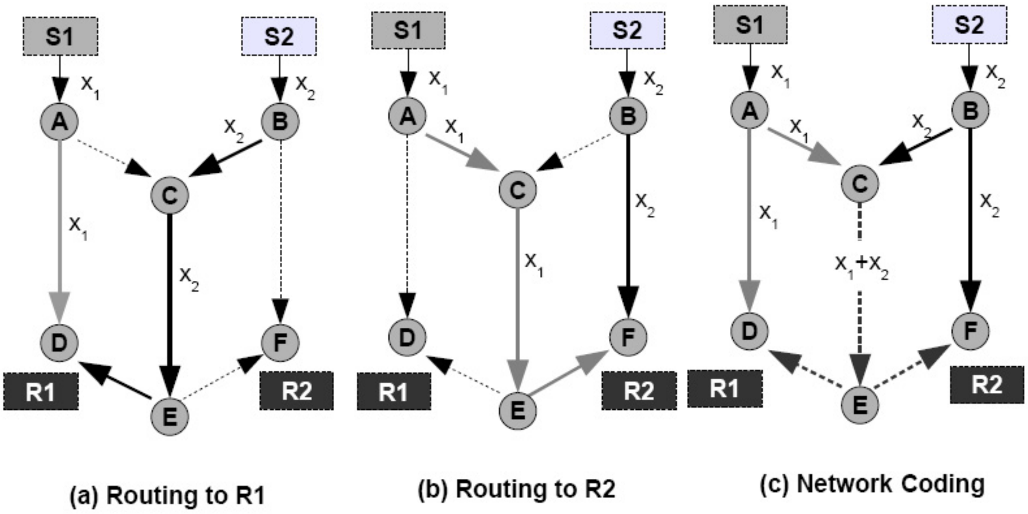


Figure 2.1: Illustration of network coding

Suppose we have a communication system operating in slotted time, with source $S1$ and $S2$ and receivers $R1$ and $R2$. At each time slot, sources $S1$ and $S2$ generate packets x_1 and x_2 respectively, both of which are to be delivered to both receivers. Under existing technologies, if receiver $R1$ uses all the network resources, it can receive both the packets assuming all links are non-interfering. The flow is x_1 through $\{AD\}$ and x_2 through $\{BC, CE, ED\}$ (see Figure 2.1(a)). Similarly, receiver $R2$ can instead receive both the packets using all network resources by receiving x_1 through $\{AC, CE, EF\}$ and x_2 through $\{BF\}$ (see Figure 2.1(b)).

However both receivers are not able to receive both the packets simultaneously due to the shared link $\{CE\}$ which can only carry either x_1 or x_2 at a time, but not both. Both x_1 and x_2 can only be simultaneously received by both receivers if node C can combine these two packets and forward it as a single packet. This is exactly what network coding does.

With network coding, packets x_1 and x_2 are $XOR - ed$ together at node C to generate an encoded packet, which is then forwarded to nodes D and F through node E as shown in Figure 2.1(c). Since node D has already received packet x_1 through $\{AD\}$, it can extract packet x_2 from the $XOR - ed$ packet by XOR-ing it again with packet x_1. Node F extracts packet x_1 with a similar operation.

At this point, we may also introduce two basic terminologies commonly used in network coding literature. Packets like packets x_1 and x_2 from above example are called ***native packets***. These basic units are combined to form an ***encoded packet***. So the $XOR - ed$ packet $x_1 \oplus x_2$ is an encoded packet.

2.1 State of the Art in Network Coding

The first work on network coding is the seminal paper by Ahlswede et al. [2], where it was shown that bandwidth can in general be saved by employing network coding. This and most other earlier work on network coding are primarily theoretical and assume multicast traffic. A few papers like [9] by Z. Li and B. Li show how multiple unicast sessions can be achieved with network coding. The first work in coding aware routing is [8], where the idea of *'coding aware routing'* is introduced, and suggested that this can improve the benefits of network coding in a wireless network. This observation is proven further using theoretical formulation for throughput with network coding in [10]. Some work has also just begun in the area of Physical/MAC layer aspects of wireless network coding [4,5,11]. The first practical implementation of wireless network coding is described in [3].

2.2 Practical Issues

This section basically explains some practical issues regarding encoding and decoding of packets using the network coding technique. For this purpose, survey from some literature will be described here. These references cover network coding in generaland wireless network in specific.

2.2.1 Encoding

Linear network coding described in [1] is different from concatenation, i.e. linear 'addition' packets. In linear network coding, original packets of length L results in encoded packet that also has size L, and in contrast to concatenation, the encoded packet by itself is not decodable.

Another important consideration is the size of packets that are to be combined together. Encoding small packets with larger ones reduce bandwidth saving [3], because shorter packets has to be padded with trailing $0s$ $[1,3]$. In addition, packets headed to the same next-hop cannot be coded together, since no node will be able to decode more than one packet at the same instance [3]. Furthermore, each intended next-hop should have all the packets other than that intended for itself. Therefore, the packets chosen to combine together has to be selected with all these considerations in mind.

The idea of *opportunistic coding*, aimed at maximizing the throughput is introduced in [3]. A node should maximize the number of packets delivered in a single transmission, while ensuring that each intended next-hop has enough information to decode its destined packet.

Data encoding in network coding is brought up in [12]. The idea is to multiply packets with a set of coefficients and then XOR them instead of just simple XOR as mentioned above. This work was specifically done in the file sharing case. For example, node A has a set of chunks $C = C_1, C_2, ..., C_n$, where chunks are defined as multiple pieces of file which is split up. When it decides to send the information to the neighbor B, it will choose a set of random coefficient $k_1, ..., k_n$. Then it multiplies C_i with k_i and XOR all the product together, i.e. $C' = k_1.C_1 \oplus k_2.C_2 \oplus ... \oplus k_n.C_n$. If node A holds n chunks, it should be able to generate n disjoint combinations, which can not be done through a simple XOR.

2.2.2 Decoding

Decoding requires solving a set of linear equations. In practice, this can be done as follows. A node stores the encoded vector it receives as well as its own original packets, row by row, in a so-called decoding matrix [1].

In practical situation which was implemented in [3] for the ad-hoc 802.11 network, the following method is used to decode a packet. A node, which is the previously traversed node of some packets, can store those packets in its buffer for a period of time. Even though those packets are not intended to it, it can use them in future for decoding purpose, i.e. to decode the desired packet from the encoded packet of transmission with network coding. This has an important role in implementing network coding as it creates a more coding opportunity.

In this implementation, each node has an outgoing queue and a buffer for transmitted packets. Suppose node $N1$ has packets P_1 P_2 to transmit, which are now in the outgoing queue. Once this node transmits any of these packets, lets say it is packet P_1, this transmitted packet (P_1) will then be stored in the buffer to be used in decoding future encoded packets that it may receive.

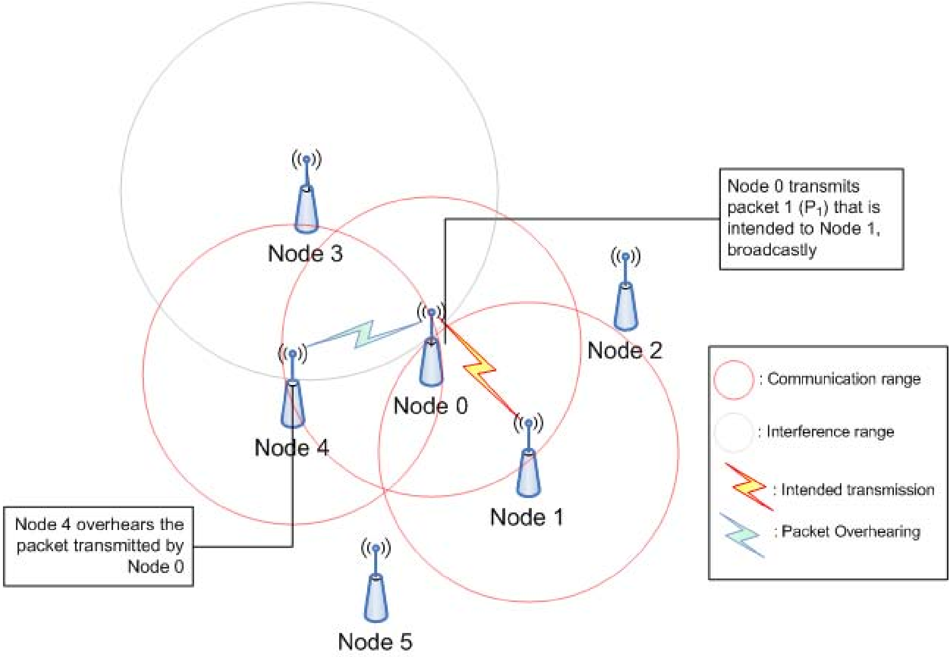


Figure 2.2: Illustration to show how 'overhearing' works

Furthermore, in [3] packet overhearing was also introduced. With packet overhearing, a packet can receive a packet without necessarily being an intended receiver of that packet. This increases the opportunity of a node to do network coding, since it has more packets to encode and decode from. Overhearing is graphically explained in Figure 2.2. Overhearing in fact exploits the broadcast nature of wireless transmission. Thus a packet transmitted by a node to one of its neighbors can be 'overheard' by other neighboring nodes as long as they are within the reception range of the transmitter node. COPE architecture which implemented opportunistic coding as well as packet overhearing is further discussed in Section 2.3.

2.3 Review of COPE architecture

The groundwork for this thesis is Coding Opportunistically (COPE), a network coding enabled wireless networking architecture proposed by Katti et. al. in [3]. In this architecture a new coding layer is inserted between the MAC and routing layer.

The underlying principles of COPE are:

- At each intermediate node outgoing packets are XORed to form a *coded packet* by combining more than one *native packets*. For each native packet, the probability that its intended next hop can decode the native packet in the coded packet is determined. If the probability is greater than a threshold, G, it is included in the coded packet.
- Each node maintains a table of its neighboring nodes and the probability these of

having its packets. This table is refreshed periodically (in this case 0.5 sec). This is done through *Opportunistic listening*, i.e. overhearing packets received by its neighbors.

- Each node has one logical queue of outgoing packets, the output queue. Virtually this is divided into 2M queues (where M is the number of neighboring nodes), with different queues for short and long packets per neighboring node.
- When coding, the first packet in the output queue **MUST** be included in the encoded packet. With this constraint, the gain is maximized by including as many native packets as possible in the encoded packet. First suitable packets with the same size is searched for, followed by packets of the different size.
- Received packets are decoded by XORing the encoded packets with other (n-1) native packets (which the receiver is expected to have).

Pseudo-Broadcast MAC

Existing IEEE 802.11 MAC protocols such CSMA/CA is not well suited to handle multi-hop communication of wireless mesh networks, one example of which is 'hidden node' problem. A transmission bottleneck is also caused by the transmissions of intermediate nodes. Moreover, the nature of network coding is to use broadcast transmission but it does not work here because of lack of back off and poor reliability.

The IEEE 802.11 MAC with unicast mode ensures reliability by retransmitting the packet for a fixed number of time until an Acknowledgment (ACK) is received. In case there is no ACK received, it is interpreted as a collision due to congestion, then the sender backs off exponentially as a congestion control technique. Hence, in the broadcast mode where there are many intended receivers, there is no ACK at all. Moreover, the collision cannot be detected by the sender, and thus does not back off.

In this paper, the problem of MAC with 'pseudo-broadcast' is solved by having the ACK functionalities in the COPE layer as well. So, one of the intended receivers of an encoded packet is set as the link layer destination (the receiver according to classical ACK), while the rest of the receivers sends ACK by piggy-backing some information using the ACK block in the COPE layer, i.e. asynchronous ACK. Furthermore, a sender schedules a retransmission for each of the native packet in the encoded packet. If any of these packets is not ACK-ed within a certain period of time, the packet is inserted at the head of the queue and retransmitted. Retransmitted packet may get encoded with other packets.

2.4 Application of Network Coding

Network coding has a lot of benefits, especially for throughput gain and robustness as well as adaptability [1]. These advantages benefit different communication scenarios in different ways. Some examples are:

2.4.1 Peer to peer file distribution

In [12], which took file sharing application in wireless mesh network case, it was shown that simple cooperation technique between nodes greatly reduces service time if network coding is applied. Moreover, the gain from network coding is higher with larger number of sources.

2.4.2 Bi-directional traffic in wireless network

With bi-directional amplification of throughput using network coding in a wireless network with bi-directional traffic a gain as high as 100% can be achieved [4].

2.4.3 Ad-hoc sensor network

Un-tuned radios in ad-hoc sensor network can operate with a very simple radio architecture where multi-hop path between information source and data sink is easily found without much overhead by employing random network coding [13].

2.5 Problems and challenges

Apart from all the benefits that have been mentioned, there remains some open research areas and challenges in network coding, especially within wireless communication field. COPE as the first implementation of network coding has been able to address the common case of unicast traffic, dynamic and potentially burst flows, and practical issues for the integration of network coding in the current network stack [3]. However, routing issue is not addressed in that implementation.

Concerning the routing issue in wireless network coding, it is suggested that further gains are possible if routing decisions are done with the awareness of coding independently [8]. In this case, there should be a mechanism to maximize the coding opportunity by routing the flows in a certain direction.

As we know, network coding does not work if the traffic flows are in the same direction. Therefore the flows should be bi-directional when network coding without packet overhearing is considered. Whereas, when packet overhearing is considered, the routing protocol should exploit as many overlap nodes as possible. Anyhow, in both cases the route might not be a shortest path any more. Such route might require more transmissions and will result in the decreasing of throughput. Furthermore, when the nodes are so much overlapped, the network will be too congested and therefore again lowering the throughput.

Complexity is another open area of concern. Implementing network coding at intermediate nodes will increase the complexity and hence cost of these nodes. The complexity of network coding has to be assessed and how the benefits weigh against the complexity cost has to be investigated.

Other challenges include integration with and deploying network coding to existing network infrastructures.

2.6 Conclusion

The important point to be highlighted here is the trade off due to the optimization for the wireless network coding. The optimization strategy can be either based on the MAC layer design, scheduling strategy, routing mechanism, or joint strategy of any of them. Furthermore, the possibility to implement those strategies in the real scenario with low complexity, thus lower cost, is another challenge for wireless communication researchers.

Chapter 3

Wireless Mesh Network and Wireless Routing

After having an in depth look at network coding, we will now discuss wireless mesh network and routing protocols for wireless mesh network. The objective of this chapter is to understand the existing technologies, which will be the basis for the work done in this project.

3.1 Wireless Mesh Network

Mobile Ad Hoc Network is an autonomous network set up by number of mobile devices communicating together over wireless links. Independent mobile devices can easily and rapidly connect together to form an ad-hoc network whenever and wherever necessary. It is a decentralized network, where all network activities are executed by the nodes themselves, for example, routing functionality is incorporated into mobile nodes [14]. The nodes dynamically organize and configure themselves to maintain connectivity throughout the ever-changing and unpredictable structure of the network.

3.1.1 Brief introduction to WMN

Wireless Mesh Network (WMN) is a relatively new key wireless networking technology that overlaps with the concept of ad-hoc network to support more sophisticated features. These sophisticated features include wireless mesh backbone and interconnection with and integration between different wireless technologies, such as Worldwide Interoperability for Microwave Access (WiMAX) and Wireless Fidelity (WiFi) [7]. These features are supported by nodes with specific functionalities, such as mesh routers and mesh gateway/bridge.

Wireless mesh networks has a wide range of interesting applications, mainly in the area of Broadband Wireless Access (BWA). One example is extending Wireless Local Area Network (WLAN) *hot-spots* to *hot-zones* [15]. Other applications include neighborhood networking [16], broadband home networking and building automation etc [7].

3.1.2 Architecture of Wireless mesh network

There are two types of nodes in a WMN, namely mesh routers and mesh clients. The mesh routers have additional functionality to support routing in mesh networks through multi-hop communication. The routers also have gateway and/or bridge functionality that supports interconnection with other networks (such as WiMAX, WiFi etc) and to the Internet. Based on the type of nodes in a WMN, the architecture can be categorized as:

Infrastructure/Backbone WMN In this type of WMN, number of mesh routers are interconnected to form wireless mesh backbone. The users (mesh clients) access the network by connecting to the backbone. External networks are also directly connected to the backbone.

Client WMN In this architecture, all nodes are mesh clients and they perform both routing and configuration, and provide end user service. The clients are connected through peer-to-peer network.

Hybrid WMN This is the combination of the previous two architectures, and is shown in 3.1.

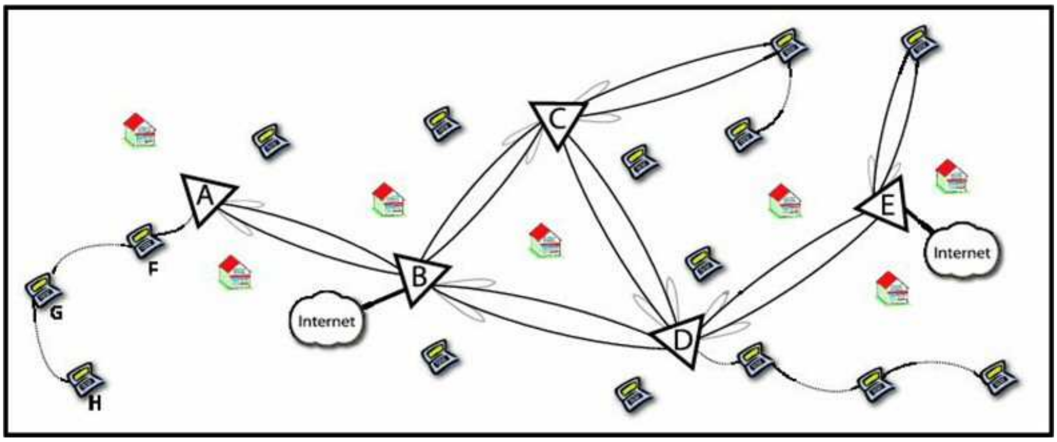


Figure 3.1: Hybrid WMN

3.1.3 Key features

The key features that distinctly mark a wireless mesh network are [7]:

Wireless backbone a sub-network of interconnected mesh routers acting as a backbone. This provide larger coverage, wider connectivity and increased robustness.

Integration Through the bridge functionality of mesh routers, network with different access technologies can be connected to each other.

Dedicated routing The mesh routers perform dedicated routing, which reduce load on end user devices, thus introducing greater flexibility and scalability.

Multiple radios The mesh routers are equipped with multiple radio to separate the routing and access functionalities. This can significantly improve the network capacity.

3.1.4 MAC for WMN

Traditional wireless MAC protocols cannot readily be used in WMN because of following differences between between the two:

- Communication in WMN is inherently multi-hop, whereas wireless MAC protocols are limited to single-hop communication. In WMN, communication at one node is affected by other nodes that are more than one hop away.
- Since there are no centralized nodes, the MAC protocol should ensure cooperation between all nodes during transmission.
- The network topology is ever-changing in a WMN although it is not as much as in ad-hoc network. Thus the MAC protocol needs to know about the current topology which can enhance cooperation between neighboring nodes and overall performance.

These issues can be addressed by either enhancing existing MAC protocols or introducing new and innovative MAC protocols. One such example is a multi-channel MAC protocol that supports communication over multiple radio using a single transceiver [17].

3.1.5 Network layer in WMN

Like most wireless networks, Internet Protocol (IP) has been accepted as the network layer protocol for wireless mesh networks. However routing protocols, which are also included in the scope of this layer, are different for WMN compared to other networks. Because of its similarity with ad-hoc networks, routing protocols designed for ad-hoc networks, such as DSR [18], DSDV routing [19] and AoDV [20] can be used for WMNs. These routing protocols are described in 3.2.

3.2 Routing protocol for Ad-Hoc network

In general, routing protocol for ad-hoc network is divided into two types of routing, i.e. reactive and pro-active. There also exist other routing protocols that are based on hierarchical, geographical, and so on. In this discussion, only first two routing protocols are discussed. We explain a few routing protocols belonging to these two category, such as DSDV which is pro-active or table driven protocol, as well as DSR and AoDV which are reactive or on-demand protocol.

3.2.1 Destination-Sequenced Distance-Vector (DSDV)

As a table driven protocol, DSDV needs to determine the routing table periodically. In case of any changes in the network, this information is broadcasted, and all host in the network run the route discovery algorithm again and store new routing information in their table [21].

In short, DSDV protocol can be described as follows: Each node in the network has a routing table which contains the next hop and the number of hops to reach each of the destinations. It also contains a sequence number which created by the destination. The sequence number is used to solve the routing loop problem [22], by selecting the route with more recent sequence number [23]. Routing information is advertised by broadcasting the packets periodically and incrementally as topological changes are detected. Concerning the route selection, if transmitter has received some new information, the latest sequence number will be used. If the sequence number is same, better metric is used.

In general, DSDV is suitable for ad-hoc network with small number of nodes. Moreover, any time when a route is needed, a route is already available in the table, thus reducing the average delay per packet. However, DSDV requires a periodic update of routing tables which implies the bandwidth efficiency reduction. It is also not suitable for very large network, since it is less scalable.

3.2.2 Dynamic Source Routing (DSR)

DSR is an on-demand routing protocol for ad-hoc network. It allow nodes to dynamically discover a source route across multiple network hops to any destination in the ad hoc network [24]. The source node includes the full route, which is complete ordered list of nodes through which the packet must pass, in the packet's header. The intermediate nodes use this to forward packets towards the destination and maintain a route cache containing routes to other nodes [25]. They can also overhear the packets and cache the routing information for future use [24].

Basically, it consists of two mechanisms which is route discovery and maintenance.

Route discovery

It is a mechanism by which the source node broadcasts a route request for a destination node. Route discovery is used only if the source node does not have a route to a specific destination in its route cache.

Route maintenance

It is a mechanism by which the source node is able to detect if the link to destination node is broken. It is used when source node is sending packets to destination. If the link is found to be broken, source can use any other route to destination that it has, otherwise it can invoke Route discovery again to find a new route.

3.2.3 Ad-Hoc On demand Distance Vector (AoDV)

In the previous section, we have seen how DSDV works. This algorithm is efficient for smaller networks.Moreover it is also not suitable for memory constrained devices [26] because of the long list of neighboring nodes that each node has to maintain. These shortcomings are addressed by the source-routing protocol Ad-Hoc On demand Distance Vector (AoDV) proposed by Perkins and Royer in [27].

AoDV is a source initiated on demand routing protocol that is specially suited for ad-hoc networks. It provides loop free routing with little overhead while maintaining the advantages of basic distance-vector routing. Because it relies very little (or at times not at all) on periodic advertisement, it is also easily scalable to large populations of mobile nodes connected through ad-hoc network.

The basic idea of AoDV is that it is a pure on-demand algorithm and route acquisition between any pair of nodes is only triggered when they have to communicate, or one of the nodes acts as an intermediary to other nodes. The nodes are aware of other nodes in its neighborhood by the use of several techniques, such as broadcast of hello messages. This optimizes response time to local movements and provide quick response time for requests for establishment of new routes.

The primary features of this algorithm are:

- Broadcast discovery packets only when necessary,
- Distinguish between local connectivity management and general topology maintenance, and
- Exchange information about changes in a node's connectivity to whichever node that may require it.

The combination of above-described features ensure the algorithm efficiently uses the bandwidth with minimum control overhead and can respond quickly to changes in the environment.

3.3 Conclusion

The working mechanism of wireless mesh network, especially ad-hoc mesh networks are presented in this chapter, along with routing techniques for wireless ad-hoc network. The following chapters represent the main work done in this project.

Chapter 4

Wireless Network Coding in Linear Topology

This chapter describes how wireless network coding works in the simple linear topology of scenario one described in Section1.2.1. The aim of this investigation is to find out the gain of network coding over conventional transmission techniques for different information flows with varying rate.

4.1 The detailed scenario

This is the most basic scenario in our investigation where wireless network coding is applied. In this case, there are n nodes arranged in a straight line. The nodes are equidistant and can only communicate with node(s) one hop away on either direction. They are also assumed to be perfectly synchronized, and transmission between nodes is perfect and without any physical layer bit error. Furthermore, there is no *time out* of any sort and packets are allowed to stay in the network as long as they are not dropped at any node due to buffer being full.

4.1.1 MAC and Routing

It is assumed that a perfectly-controlled Medium Access Control (MAC) mechanism is in place, and only non-interfering nodes transmit at any given time slot. In this way, each node can either receive a packet from its left, receive a packet from its right (for the two edge nodes, one of these reception is idle) or transmit (broadcast to both left and right nodes), as shown in Figure 4.1. We also assume that the sender node perfectly receives acknowledgment within the transmitting time slot.

Since the nodes are arranged in a straight line, there is only one possible route between any given pair of nodes. This leaves us with the following simple routing shown in Figure 4.2.

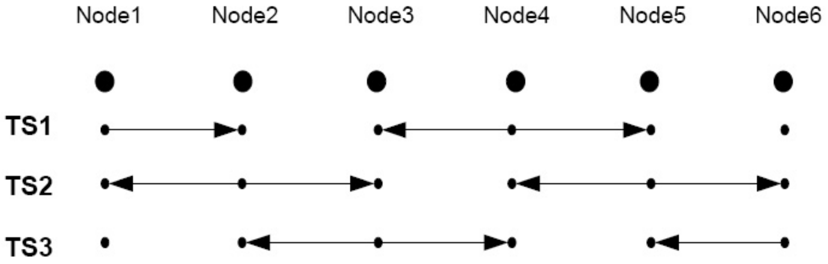


Figure 4.1: Medium Access Control in the linear node structure

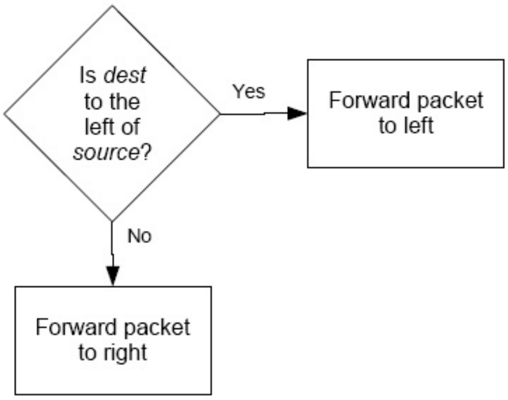


Figure 4.2: Routing mechanism for linear network structure

4.1.2 Node and Traffic Packet properties

In this study, we assume that all generated packets are of same size and type. In both scenarios, the packet arrival is modeled probabilistically as well as deterministically. When probabilistic, packet arrival is modeled as an ON/OFF binomial process with packet arrival (ON) probability given by the offered load on the given flow. In the deterministic case, a packet is introduced in the flow every t_{th} time slot, where $\frac{1}{t}$ is the offered load for that flow.

In both cases, packet arrival at any node is independent of that in other nodes, but the process is identical at each node.

The queue at each node is First In First Out (FIFO). In order to simplify the process at source nodes, we do not consider two separate queues for generated and forwarded packets. So packets generated at the source nodes are placed at the end of the ’outgoing packet queue’. Therefore, a queue scheduling is not considered.

4.1.3 Implementation strategy

At each time slot, we will perform three processes, namely *Packet generation*, *Transmission* and *Reception*, which are described below:

Packet generation

The packet is generated at nodes that are the sources of any of the given flows. A node can only be source to one flow, though it can be a destination to multiple flows. Each generated packet is added at the end of outgoing queue of that node. As stated in Chapter 1, the flows are fixed for each simulation.

Each packet has the following header information:

- Source address
- Destination address
- Time stamp indicating the time of packet generation
- Next hop

Transmission

This transmission function traverses through the set of nodes checking for nodes with available packets for transmission in the outgoing queue (only nodes with right to transmit will be checked). All packets that are to be transmitted are marked and then passed to the *reception* function.

Reception

After transmission, the nodes are then checked whether they are receiving any packets. For each of the receiving node, the following steps shown in Figure 4.3 are performed.

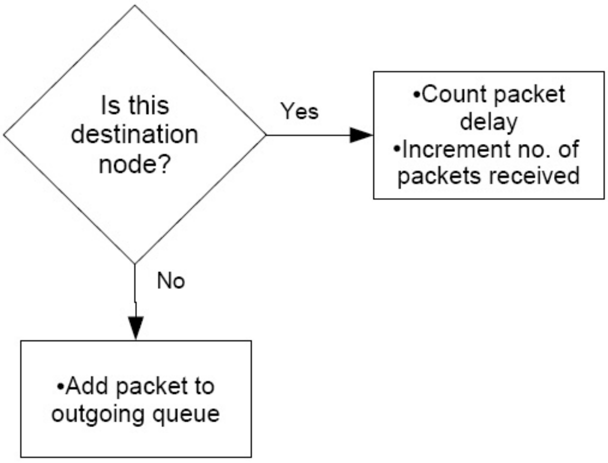


Figure 4.3: Flowchart for decision on packets received at receiving node

Detailed implementation of network coding in linear topology

The implementation of this first scenario, i.e. network coding in linear topology is quite simple because of the network architecture. Flowchart for the implemented network coding algorithm in linear topology is shown in Figure 4.4.

4.2 Analytical Discussion

Theoretically, the supportable rate in a network depends on the transmission handling capacity of associated network devices, in this case the nodes. In this scenario, each node gets to transmit once every three time slots, so the maximum supportable data rate in the conventional case, i.e. without network coding, is $\frac{1}{3}$.

However with network coding in linear architecture, a node can combine two native packets to form an encoded packet, provided both destinations have the other native packet. This is because there are two possible directions for a node in this linear topology and so the maximum number of packets transmitted in a single transmission is 2. Therefore the maximum achievable throughput is twice of the conventional case, i.e. $\frac{2}{3}$. However the rate from one direction should not be more than $\frac{1}{3}$ as this is the limit without network coding, and with network coding multiple native packets from flows in the same direction cannot be encoded together.

Theoretically speaking, these maximum supportable rates for both cases are the offered rates beyond which packet delay starts to go towards infinity.

Having the condition of the supportable rate, especially in the network coding case, it is interesting to see how the asymmetric traffic gives the impact to the system throughput. As mentioned earlier, the maximum supportable rate in one direction for the network coding is $\frac{1}{3}$. If the arrival rate from one direction is less than $\frac{1}{3}$ and other is $\frac{1}{3}$ or even more, the system will not be able to achieve the maximum throughput.

4.2.1 Definition of key terms

There are some key terms related to network coding that should be defined clearly before discussing the analytical result.

Merger nodes

The transmitting nodes through which two or more flows overlap are called merger nodes. These nodes transmit packets from all the overlapping flows. If at least two of the overlapping flows are in opposite direction, then there arises opportunity to do network coding at these nodes.

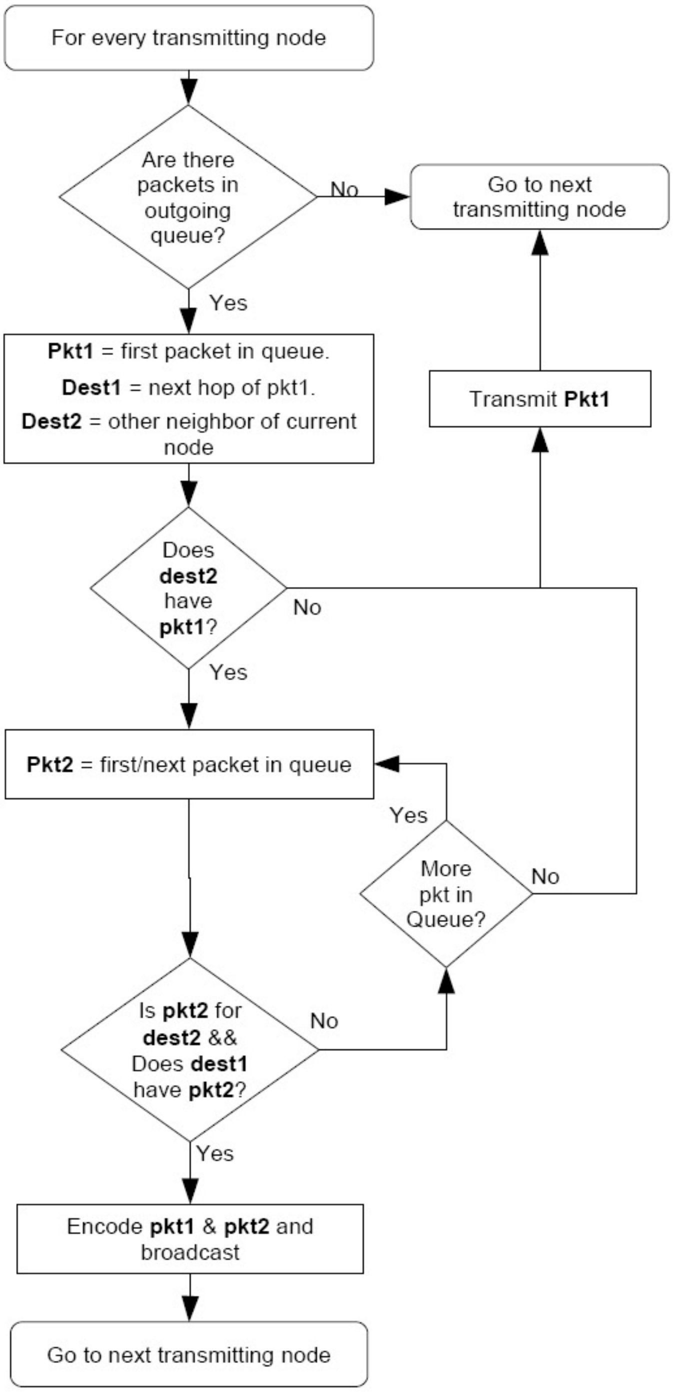


Figure 4.4: Implementation of network coding

Coding opportunity

Coding opportunity is a measure of the 'degree' of network coding in a network, and ranges from 0 (no coding) to 1 (100% coding opportunity). In this linear architecture, it is calculated as

$$C.O = 1 - \frac{|\sum\ rates\ of\ flow\ in\ one\ direction\ -\ \sum rates\ of\ flow\ in\ opposite\ direction|}{Max(\sum\ rates\ of\ flow\ in\ each\ direction)} \tag{4.1}$$

where the rates are taken at each transmitting node in the network.

Throughput gain due to network coding will be directly proportional to the coding opportunity, with coding opportunity of 1 yielding the highest gain (which is around 33% for decode and forward [3] case).

It should however be noted that it is the merger nodes that impact the throughput gain. For example, in the case of flows $[1-8]$ and $[5-4]$, node 5 is the merger node and thus there will be some throughput gain. On the other hand, for flows $[1-5]$ and $[8-4]$ there are no merger nodes and hence no benefit from network coding.

4.2.2 Analytical maximum supportable rate

For a linear structure, given a set of flows and corresponding packet arrival rates, we can determine if this set can be supported with tolerable delay in presence of networking coding. For this set to be supported, following two conditions has to be fulfilled.

$$Max(\sum rates\ of\ flow\ in\ any\ one\ direction) \leq \frac{1}{3} \tag{4.2}$$

$$\frac{\sum(rates\ of\ all\ flow)}{1\ +\ C.O.} \leq \frac{1}{3} \tag{4.3}$$

Where the rates are taken at each transmitting node in the network.

Based on the above Equation 4.2 and Equation 4.3, we can analytically determine the supportability of any given set of flows, and corresponding rates. This supportable rate region is shown in Figure 4.5 below, where the two opposite flow directions are represented by $R1$ and $R2$

4.3 Numerical results and discussion

The results presented in Section 4.2 are theoretical bounds on the performance of network coding in a linear structure. In this section, we present simulation results to verify the analytical model.

4.3.1 System setup and Investigation objective

The basic simulation system setup for this linear structure is an array of six nodes equidistant from each other. We assume only neighboring nodes can communicate and interfere

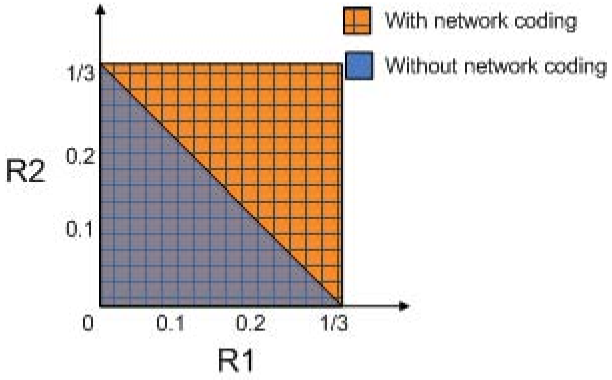


Figure 4.5: Supportable rate region with and without network coding

with each other. Flows in the network can be in two possible directions, either from left to right or vice versa; with only one possible route between any two given nodes. Packets are generated at the source node according to the flow rate in the deterministic manner described in Section 4.1.2.

In this numerical study, we will investigate the impact of following on network coding:

- Offered load,
- Interaction between flows, and
- Traffic asymmetry.

Offered rate

One important factor to see the gain of network coding over conventional transmission is the offered rate in the network. The offered rate is defined as mean number of packets generated per time slot.

Interaction between flows

Another interesting study is the impact of different combination of flow on the performance of both conventional transmission and with network coding. For the network coding case, we defined the term *Merger Node* which is the node that can transmit packets towards multiple destinations in a single time slot. On the other hand, in the case of without network coding there is no such option.

Traffic Asymmetry

The last investigation is to observe how network coding performs in asymmetric traffic condition, where offered load for the bi-directional flows are not the same.

4.3.2 Performance measure

For all the above test cases, an important performance measurement metric is the system throughput. However this alone does not clearly show all aspect of system performance and so other measurement metrics are also required to evaluate the performance. In this study, we also consider the following:

- Transmissions per received packet - Number of transmissions required to deliver a packet has its implication on transmission power requirement at the node, which is an important design parameter in wireless mesh networks.
- Delay (in time slots) per received packet - Delay is the total number of time slots a packet need to reach its destination node.

4.3.3 Investigation on offered rate

In this test scenario we have two flows from the opposite direction. The first flow is from node one to node six, while the other has its source at node six and ends at node one. The traffic for both these flows is identical and is varied as follows:
$\{1/20, 1/15, 1/10, 1/6, 1/5, 1/4, 1/3, 1/2\}$.
Figures 4.6 and 4.7 show the throughput and delay performance of the network coding and conventional case for this scenario.

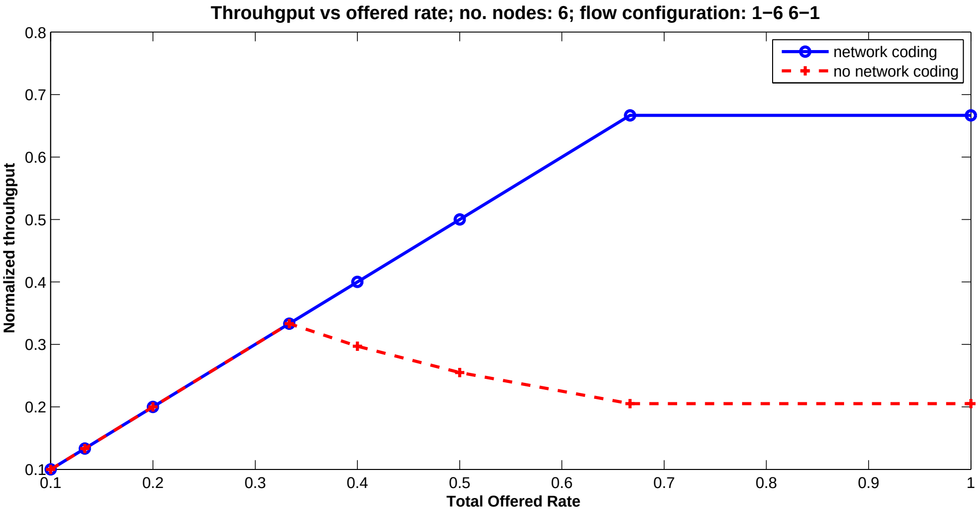


Figure 4.6: Throuhgput performance with 1-6 and 6-1 flow configuration

From Figure 4.6, we can see that in both cases the throughput increases linearly with the offered rate for rates up to $\frac{1}{3}$. Beyond that, network coding continues to increase at the same rate until offered rate of $\frac{2}{3}$, after which it becomes saturated, and remains constant at $\frac{2}{3}$ though the rate is increased further. On the other hand, throughput of the conventional case falls up to offered rate of $\frac{2}{3}$, after which it becomes constant. The fall in the region

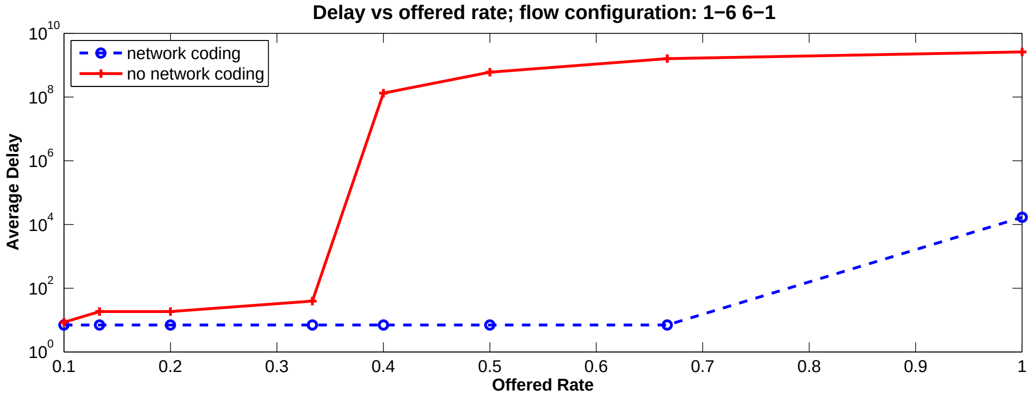


Figure 4.7: Delay performance with 1-6 and 6-1 flow configuration

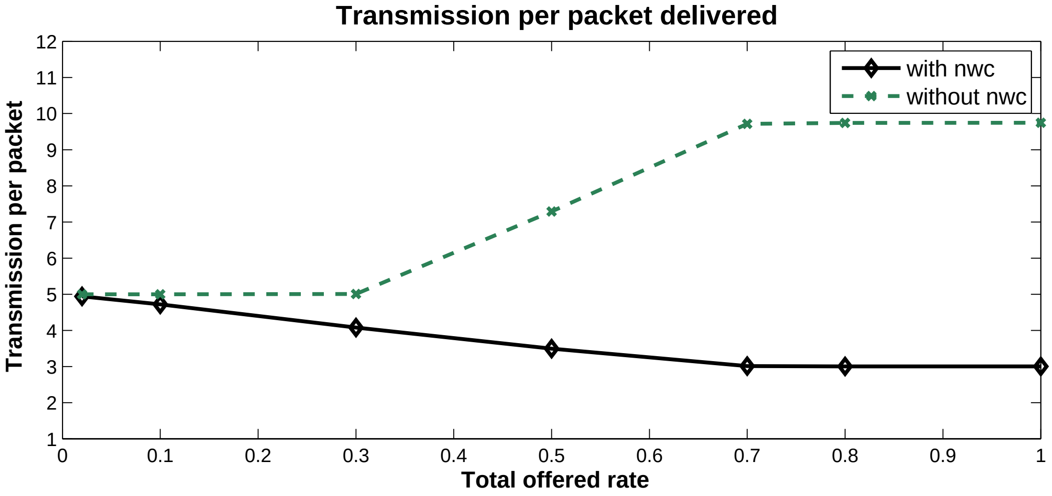


Figure 4.8: Transmissions per delivered packet for 1-6 and 6-1 flow with and without network coding

between offered load of $\frac{1}{3}$ and $\frac{2}{3}$ is because, here the load in each direction is still less than the maximum supportable rate of $\frac{1}{3}$ and so congestion in the network continues to build up. Beyond this region network becomes fully congested, and hence no variation in throughput.

Thus we can observe the benefit of network coding for the region of offered rate between $\frac{1}{3}$ and $\frac{2}{3}$. In this region multiple information flows are combined to support offered load that could not be independently supported. This confirms the theoretical bounds presented in Section 4.2 and also that illustrated through the butterfly example of Figure 2.1 in Chapter 2. However for both with and without network coding, when offered rate exceeds the supportable rate region, which is $\frac{1}{3}$ for without network coding and $\frac{2}{3}$ for that with network coding, we observe that the delay shoots up to a very high value. In fact, theoretically this is infinity. We can observe this from Figure 4.7.

We further observe that for offered rate region below $\frac{1}{3}$, there is no throughput benefit from network coding. This may give the false impression that network coding doesnt give any benefit at all in this region, which is not the case. We can see from Figure 4.8, that though the same number of packets are delivered with and without network coding, network coding achieves it with less umber of transmissions.

4.3.4 Investigation on interaction between flows or merger nodes

Merger node, which is a node that is transmitting at least a couple of bi-directional flow or more, is introduced in Section 4.2. We would like to observe how the system performs when the flows are of different length, i.e. the number of merger nodes in the flow varies. Throughput with network coding and without network coding under different combination of merger nodes is given in Figure 4.9. Like the previous test case, the bidirectional traffic is identical in this case too.

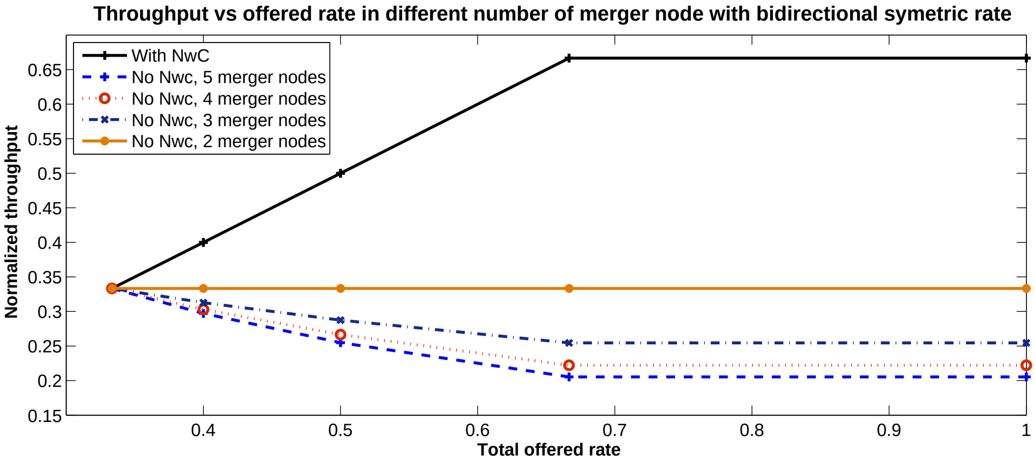


Figure 4.9: Throughput performance vs offered rate in different number of overlapping link with symmetric bidirectional traffic

First of all, throughput with network coding is not affected by the number of merger nodes, and so we just show one curve. However, for the conventional case the throughput falls between the offered rate region $\frac{1}{3}$ and $\frac{2}{3}$ with increased number of merger nodes, and then becomes constant after offered rate of $\frac{2}{3}$. This fall in throughput beyond supportable rate region for the convensional case is due to increasing congestion with increasing offered rate. The delay behavior is similar to that presented for previous test case, and thus is not produced here.

4.3.5 Investigation on traffic asymmetry

Our final investigation is about the impact of traffic assymetry on network coding. As earlier, we assume two flows with the following offered rates respectively: {[0.05 0.1], [0.1 0.2], [0.1 0.3], [0.1 0.4] and [0.1 0.5]}. Throughput results for this setup is shown in Figure 4.10 and compared with that of a symmetric traffic case.

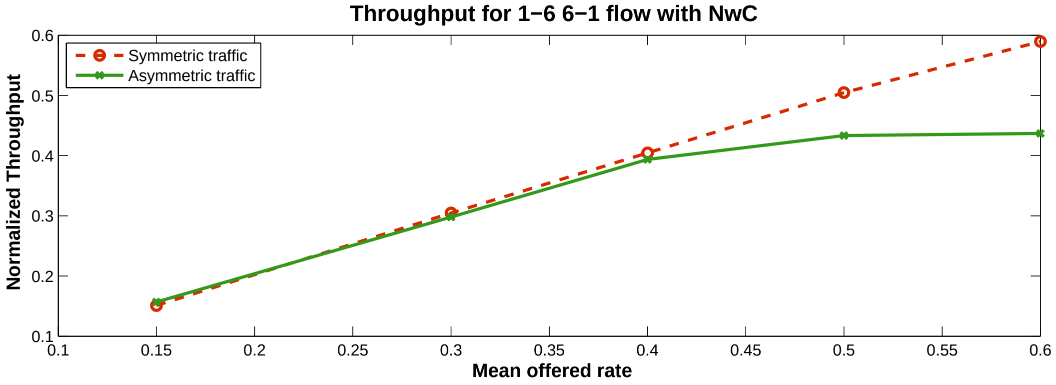


Figure 4.10: Throughput performance of asymmetric and symmetric traffic

It is evident from Figure 4.10 that the asymmetric traffic impacts the performance of network coding. As we can observe from the asymmetric traffic case, only the lower offered load is absorbed by the higher load. Between the last two offered load, i.e. {[0.1 0.4] and [0.1 0.5]} there is no throughput gain, though the sum of these two offered load is less than the maximum supportable rate of $\frac{2}{3}$. This pertains to the observation made in Section 4.2 that maximum rate of each flow should also be less than $\frac{1}{3}$ to be supported by network coding.

This result proves one important characteristic of network coding, which is the requirement of available bi-directional information flow to benefit from network coding.

4.4 Conclusion

From the above analytical study and simulation results, we conclude that network coding significantly improves the performance of a wireless network having a linear topology.

The supportable rate region ranges from the maximum supportable rate for conventional case (total $\frac{1}{3}$) to the maximum supportable rate for network coding ($\frac{1}{3}$ for each direction of flow), depending on information flows and traffic symmetry.

Asymmetric traffic from opposite direction has some impact on throughput performance of the network coding as well. The impact is the reduction of throughput gain of the system due to network coding.

Chapter 5

Wireless network coding in a grid structure

In the previous chapter, we analyzed wireless network coding in single dimension linear network. Here we will move to the two-dimensional grid architecture, where both routing and scheduling has to be optimized for network coding. This is the scenario two introduced earlier in Section 1.2.

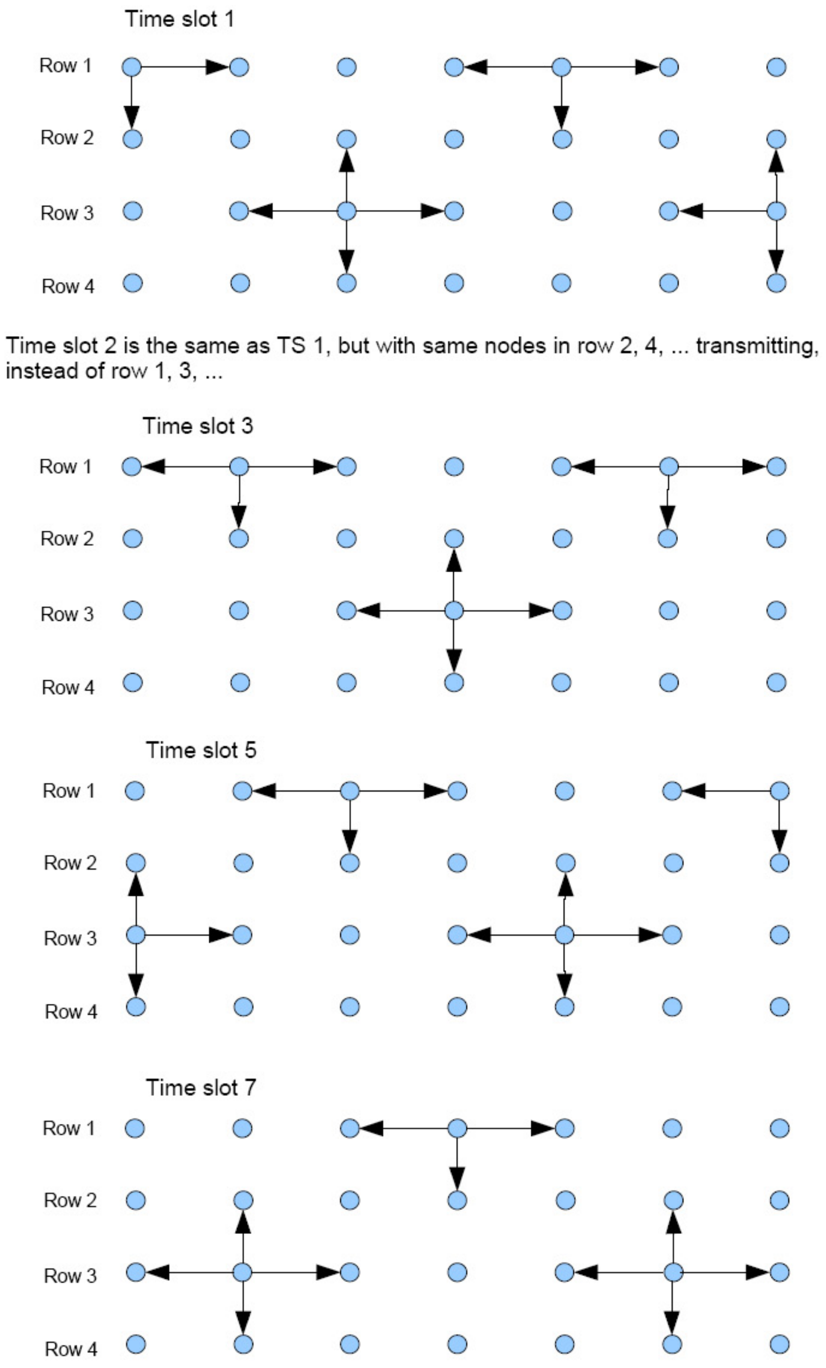
The system design setup is a collection of $N \times M$ nodes arranged in a grid structure. All nodes are equidistant and can transmit to its adjacent nodes, while transmission from nodes two hops apart interfere with each other. Flows in the network are identified by source node, destination node and packet arrival rate or flow rate. Finding a suitable route from source to destination that maximizes throughput using network coding for a given set of flows is the primary objective of this part of the study.

5.1 Scheduling in grid structure

Scheduling in this grid structure is TDMA-like and the turn to transmit for each node is evenly rotated among the set of nodes. It is assumed that nodes one hop away can communicate with each other, while those two hops apart can only 'hear' and hence interfere with each other. With this assumption the chance to transmit for a node is shared with seven other nodes. So, each node gets to transmit once every eight time slots. This scheduling method is shown in Figure 5.1. Here we assume that there is perfect synchronization among the nodes and the scheduling is controlled by an authoritative central server in an orderly manner.

5.2 Routing in Grid structure

Unlike linear structure, in a grid, there are many possible routes from source node to destination node, even if only the shortest path is considered. Moreover, wireless network coding can only benefit for the case where there is bi-directional information flow. This



Time slot 2 is the same as TS 1, but with same nodes in row 2, 4, ... transmitting, instead of row 1, 3, ...

Figure 5.1: Node scheduling in grid structure

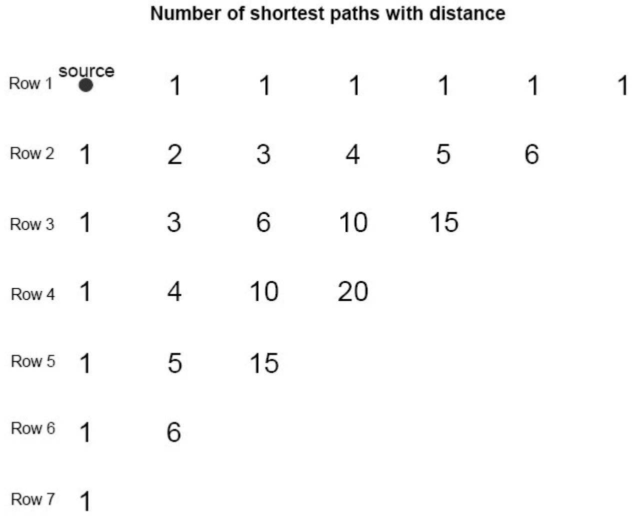
entails a well designed routing algorithm that will balance between minimizing required transmission and maximizing gain from wireless network coding.

5.2.1 Routing without network coding

With no wireless network coding, the routing objective is to choose route with shortest path. There can be more than one shortest path from a given source node to its destination node. In our case, we get the set of all shortest paths and randomly choose one of them as the route.

The first step in selecting a route is to determine how many shortest paths there are between source and destination node. There is a relation between the relative position of source and destination and the number of shortest path, and as we can see from Figure 5.2, it follows the Pascal's Triangle. The row of the triangle is the distance between source and destination, while position in that row is given by the vertical separation, and is found using the following:

$$|x\, position\, of\, source\ -\ x\, position\, of\, destination|\ +\ 1$$



Number of shortest paths with distance

Row 1	source ●	1	1	1	1	1	1
Row 2	1	2	3	4	5	6	
Row 3	1	3	6	10	15		
Row 4	1	4	10	20			
Row 5	1	5	15				
Row 6	1	6					
Row 7	1						

Figure 5.2: Relation between node position and number of shortest paths

The next step is to find the set of shortest paths. We have implemented this as a recursive algorithm as given by the flowchart in Figure 5.3.

Once the set of all shortest paths are found using the above algorithm, one of them is randomly chosen as the route between the given two nodes.

5.2.2 Routing with network coding

With wireless network coding, the situation is not as straight forward as that described in Section 5.2.1 and routes have to be chosen such that there is as much interaction as possi-

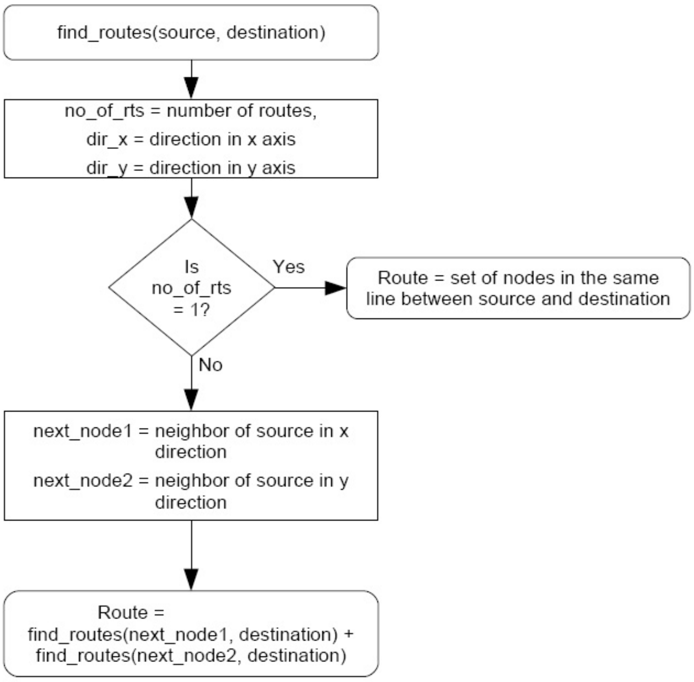


Figure 5.3: Flowchart for algorithm to find the set of shortest paths between a given source and destination

ble between the flows to promote network coding and simultaneously avoid unnecessary congestion. There are no existing routing algorithms that operate with this objective, and so in this project we propose two such algorithms described below.

Highway based routing with wireless network coding

In this novel approach, a collection of adjacent nodes across the network are selected as *'routing highways'* to be marked as preferred route path. The steps of this algorithm are as follows:

1. Each source node chooses the *highway entry node* which is a node on the *routing highway* that is closest to the source node. In the case of more than one option, *highway entry node* is randomly chosen from the possible nodes. A *highway exit node* is also chosen similarly, but in this case it is ensured that this *entry node* is closest to both, the destination as well as the *highway entry node*.
2. Route from source to *highway entry node* is the shortest path between these two points as found according to the algorithm described in Section 5.2.1. Similar is the case with the route between *highway exit node* and destination.
3. The route between *highway entry node* and *highway exit node* is the path between these points through the network of *'routing highway'*. Once again, if there are more than

one option, one is chosen randomly.

4. For each flow, the route is set up for the first packet and that is used for all subsequent packets.
5. The reason for preselecting certain *'routing highways'* is to force routes to overlap with each other as much as possible and thus facilitate network coding.

An example of routing highway is shown in Figure 5.4 below.

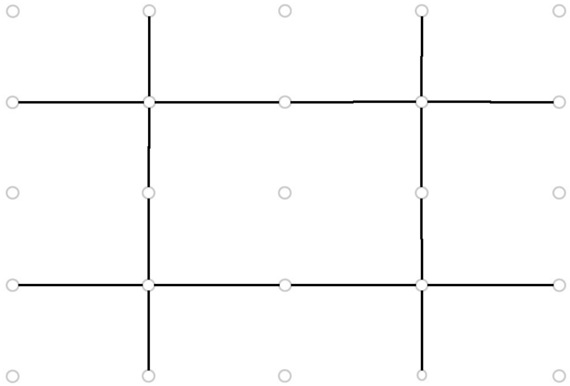


Figure 5.4: An example of routing highway

Adaptable routing algorithm for wireless network coding

This is another routing algorithm that is also aimed at maximizing the coding opportunity in a network. In this case, each flow is routed with the aim to optimize routing with network coding considering existing flows in the network.

This algorithm works as follows:

1. Route for first flow is selected according to shortest path algorithm as described in Section 5.2.1.
2. For each subsequent flows, a shortest path route is chosen such that it has maximum number of nodes that is carrying a node in direction opposite to existing flows.
3. However if the total load on a route with bi-directional traffic already exceeds the capacity, or if there are no nodes with possible bi-directional traffic, then route for new flow is chosen to avoid overlapping with existing flows as much as possible. This increases the network coding opportunity for future flows and minimizes congestion.

An example of this algorithm with graphical description is presented in Figure 5.5 below.

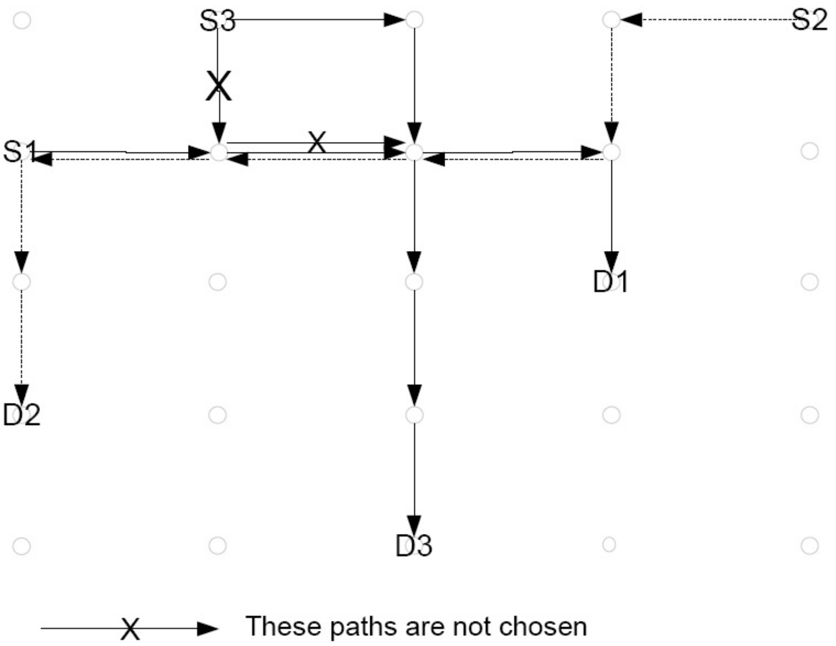


Figure 5.5: Example of Adaptable routing algorithm for Wireless Network Coding

5.3 Analytical finding on supportable rate region

Before implementing the two routing algorithms proposed above, we would like to carry out a theoretical analysis on throughput benefits of network coding in this grid architecture. The main objective of this analytical study is to find the supportable rate region when network coding is used in the network. We study two scenarios, corresponding to the two extreme possibilities presented in Figure 5.6 and Figure 5.9, and compare that with no network coding case. The first scenario is when flows in horizontal direction are completely separated from flows in vertical direction, and so at most only two packets can be combined by network coding. In the latter scenario, we assume perfect overhearing during transmission so that all nodes within the communication region of a transmitting node will get a copy of the transmitted packet. This enables a maximum of four packets to be combined together into an encoded packet, which is the maximum possible in grid structure.

In this study we assume that packet arrival is a deterministic process.

Terminology

First we introduce the different terminologies used in this analysis. These are illustrated in Figure 5.6.

The rates R_{1+} and R_{1-} refer to the sum of rates of horizontal flows in the left to right and right to left direction respectively. Similarly, rates R_{2+} and R_{2-} refer to the sum of rates of vertical flows in the top to bottom and bottom to top direction respectively.

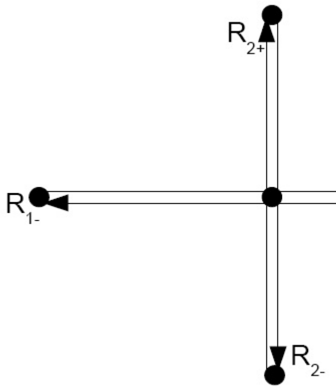


Figure 5.6: Scenario of theoretical analysis in supportable rate region for grid structure

5.3.1 Supportable rate region without network coding

From Section 5.1, we learn that a given node can at most support a rate of $\frac{1}{8}$. When network coding is not employed no packets can be combined at the node, and so the sum of the rates of all flows should not exceed the supportable rate of $\frac{1}{8}$. This can be expressed by the following condition of Equations 5.1. The supportable region is graphically presented in Figure 5.7

$$\begin{aligned} R_1 &\leq \frac{1}{8} \\ R_2 &\leq \frac{1}{8} \\ \sum(R_1, R_2) &\leq \frac{1}{8} \end{aligned} \tag{5.1}$$

where

$$\begin{aligned} R_1 &= \sum(R_{1+}, R_{1-}) \\ R_2 &= \sum(R_{2+}, R_{2-}) \\ R_{1+}, R_{1-}, R_{2+}, R_{2-} &\geq 0 \end{aligned}$$

5.3.2 Supportable rate region with network coding, but without overhearing

In this scenario, we assume that all horizontal flows are strictly separated from the vertical flows. Thus, the node can only encode packets from either horizontal or vertical flows, but not both simultaneously. Therefore, rates R_{1+} and R_{1-} can absorb each other, as long as none of them are individually more than $\frac{1}{8}$. Similarly rates R_{2+} and R_{2-} can also absorb each other, as long as they are individually not more than $\frac{1}{8}$. However since horizontal

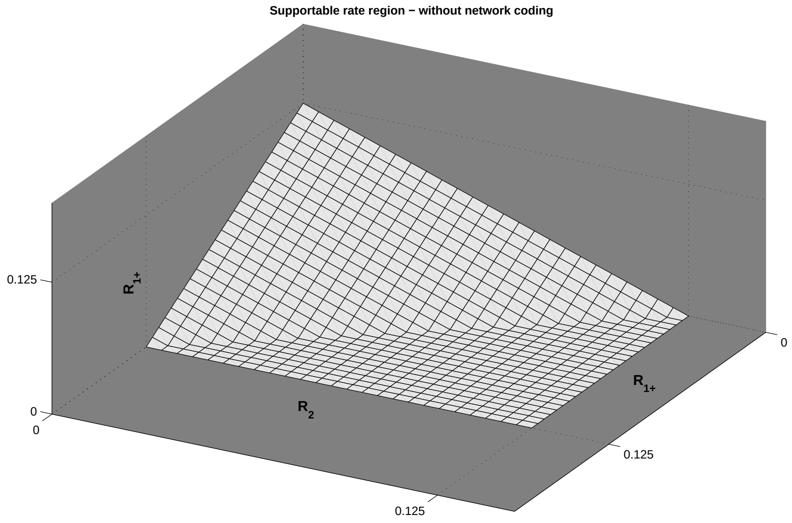


Figure 5.7: Supportable rate region without network coding

flows do not interact with vertical flows, rates $R_{1max} = max(R_{1+}, R_{1-})$ and $R_{2max} = max(R_{2+}, R_{2-})$ cannot absorb each other, and so the sum of R_{1max} and R_{2max} has to be less than or equal to $\frac{1}{8}$. This results in a maximum achievable throughput of $2 \times \frac{1}{8}$ or $\frac{1}{4}$.

Using the notations above, we can express the supportable rate region for this scenario as expressed in Equation 5.2 and shown in Figure 5.8.

$$\begin{aligned} R_{1max} &\leq \frac{1}{8} \\ R_{2max} &\leq \frac{1}{8} \\ \sum(R_{1max}, R_{2max}) &\leq \frac{1}{8} \end{aligned} \tag{5.2}$$

5.3.3 Supportable rate region with network coding and overhearing

The last analysis scenario is the case of perfect overhearing depicted in Figure 5.9. Here we assume that nodes within communication distance can overhear transmitted packet, and thus facilitating up to the maximum encoding of four packets.

In this case, the conditions of R_{1max} and R_{2max} of Equation 5.2 stands valid, and in addition rates R_{1max} and R_{2max} can also absorb each other provided they are not individually more than $\frac{1}{8}$. Therefore, we get a maximum achievable throughput of $4 \times \frac{1}{8}$ or $\frac{1}{2}$. These conditions are expressed numerically in Equation 5.3 and depicted in Figure 5.10.

$$max(R_{1max}, R_{2max}) \leq \frac{1}{8} \tag{5.3}$$

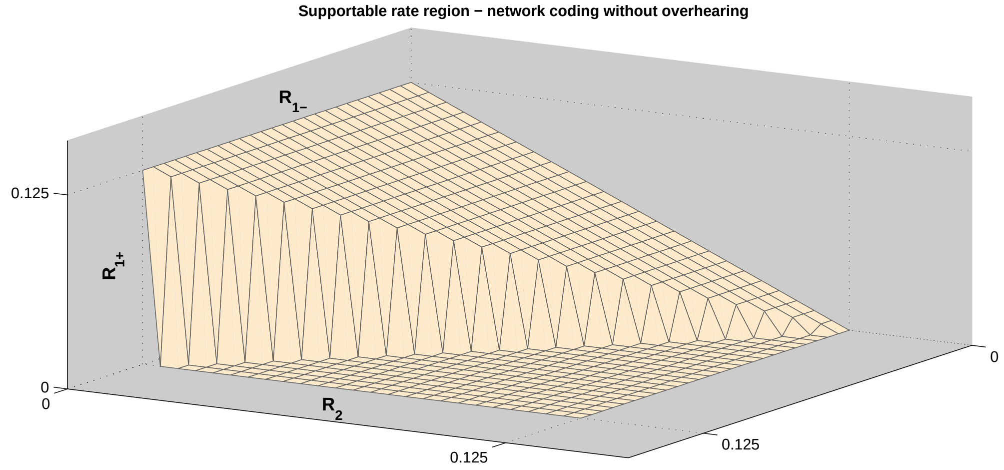


Figure 5.8: Supportable rate region with network coding, but no overhearing

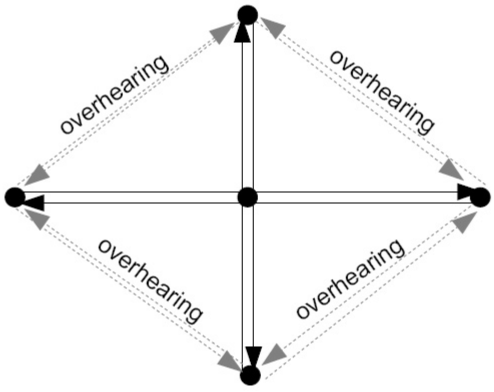


Figure 5.9: Scenario with complete overhearing

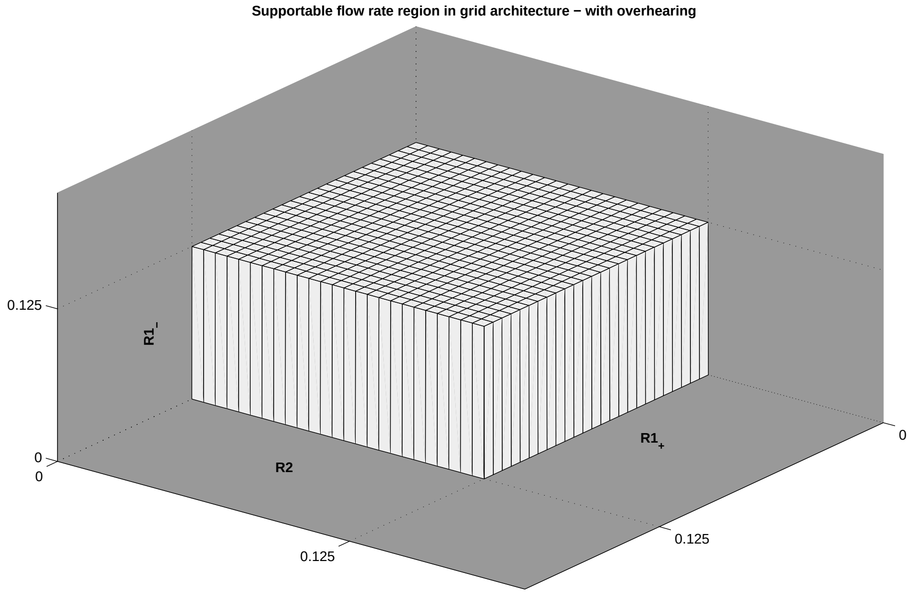


Figure 5.10: Supportable rate region with network coding and overhearing

5.4 Numerical results and discussion

Having described in detail the proposed algorithms, in this section we present the test setup used to evaluate their performance and analysis of the numerical results obtained.

5.4.1 Test setup

The simulation setup is a 5×5 grid network with 25 nodes. Results are presented for different number of flows representing different network condition (sparse traffic, congested traffic etc.), with both identical and non-identical traffic rates among the flows. The different combination of source and destination node pairs are:
{[2 18] [21 4] [6 15] [15 16] [5 11] [24 3] [20 8] [1 10] [22 20] [19 5]}
For identical flow traffic, offered traffic rate per flow is from 0.01 to 0.2, while in the non-identical case, they are chosen from -20% to +20% across different flows around these values.

5.4.2 Test cases - Routing algorithms with wireless network coding

Intuitively we can predict that performance of above described algorithms will depend to a great extent on the information flow and offered rate of the each flow in the network. Moreover, with routing highway approach how the *'routing highways'* are chosen will greatly influence the performance. Obviously, route chosen through these *'routing* *highways'* may not always be the shortest path. So when there are only few flows in the network, cost of taking a longer route may outweigh the benefits of network coding in terms of throughput.

Considering all these, we outline the following test cases for investigation.

- Impact of the number of flows in the network
- Impact of traffic intesity (flow rate)
- Impact of traffic asymmetry
- Impact of choice of *'routing highway'*

The performance measurement metric is the same as that for linear structure as given in Section 4.3.2.

5.4.3 Results and Discussions on finding: Routing Highway based algorithm

In this section we present results of above test cases and discuss the findings in light of analytical results presented in Section 5.3. All simulations are done with the highway shown in Figure 5.4 under identical traffic conditions, unless otherwise mentioned.

Impact of the number of flows and traffic rate in flow

Impact of the number of flows in the network and offered flow traffic rate is investigated in terms of throughput and average delay per packet for different flows with varying offered load for each flow. Results with network coding are compared to those without network coding to observe the gain with network coding. These results are presented in Figure 5.11.

The most obvious observation in Figure 5.11, is the clear throughput gain due to network coding which ranges from around 100% to almost 300%. We can observe that all curves are divided into two regions, namely below offered rate of 0.125 (or $\frac{1}{8}$, which is the capacity of each node without network coding) and above it. The throughput is found to be constant for all curves when the offered rate is more than $\frac{1}{8}$.

Without network coding, the maximum achieved throughput in saturation range is $\frac{1}{8}$, which supports the analytical hypothesis of Section 5.3. This happens for higher number of flows (six and 10) because here most nodes are 'saturated' with various flows. In the first region (below offered rate of 0.125), we find that achieved throughput is higher than node capacity of 0.125. This is possible because throughput of non-overlapping flows add up. As expected, average delay per packet shoots up with increasing load, but at different rates. With 10 flows, delay per packet is almost twenty thousand time slots for an offered load as low as 0.05, whereas this level is crossed with other flows for more than twice that offered load.

One interesting observation with network coding is that throughput increases with increasing flows up to certain level and then falls. Here we see that the highest throughput

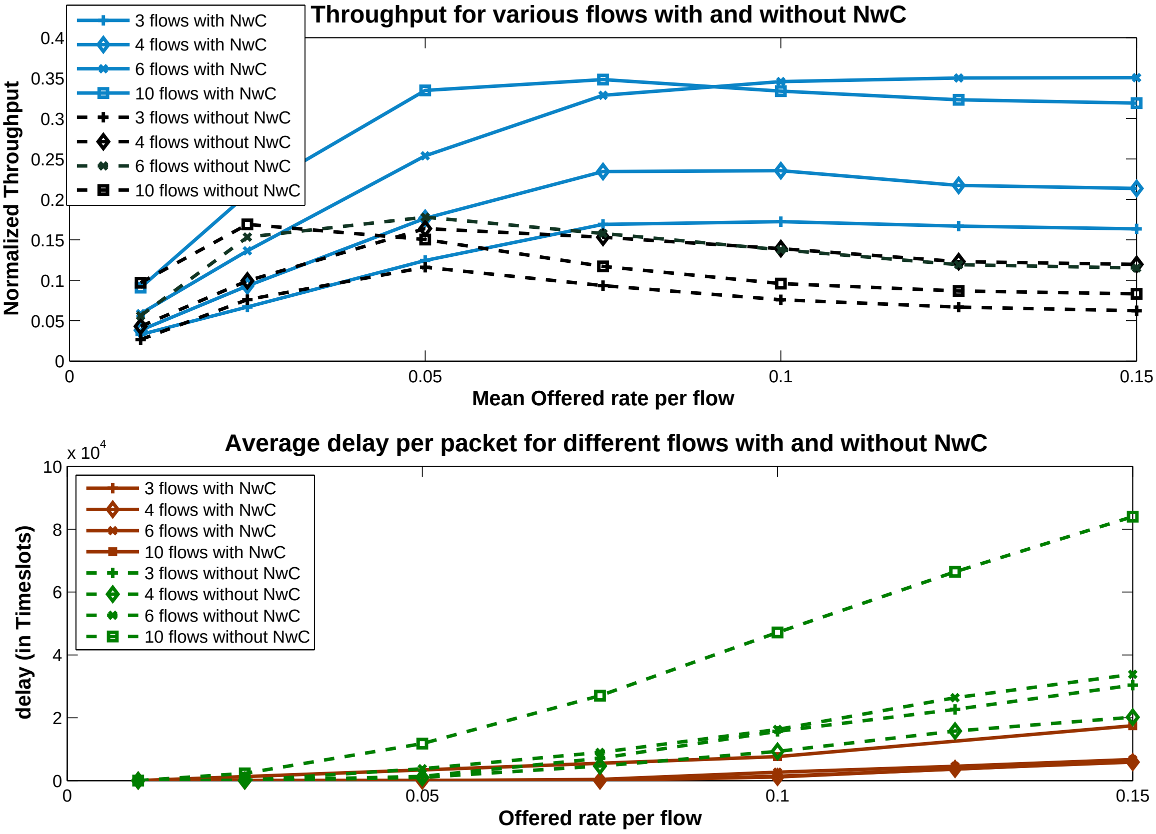


Figure 5.11: Throughput and Delay for different flow and offered traffic with and without network coding

of around 0.35 packets per time slot is obtained for six flows, but with ten flows throughput falls for higher offered load. This is because, with routing highways most routes pass through the intersecting nodes (i.e. nodes at which more than *routing highways* cross), making them more congested compared to other nodes. So, these intersecting nodes become bottle neck for the highway, resulting in fall in throughput.

Choice of *'Routing Highway'*

The choice of *'Routing Highway'* can have an impact on the performance of network coding. To analyze this impact, we compare the performance of highway shown in Figure 5.4 - termed highway one - with that shown in Figure 5.12 - termed highway two. The results are presented in Figure 5.13.

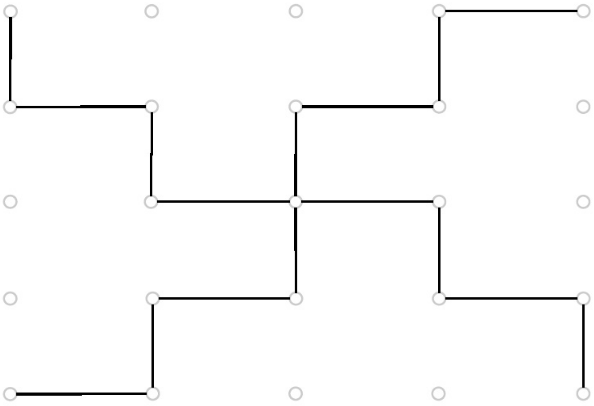


Figure 5.12: Second 'routing highway' design

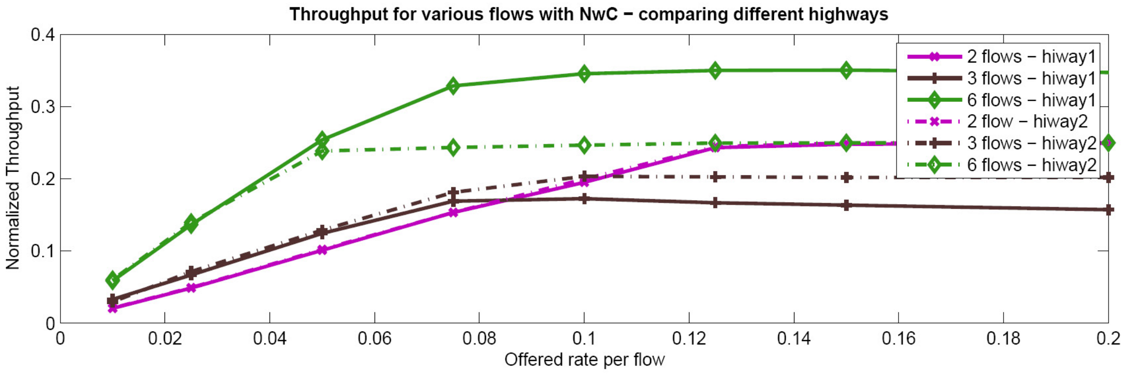


Figure 5.13: Comparison between two different 'routing highway' design

First of all, we must acknowledge the main difference between highway one and highway two structure, which is the number of *routing highways*. The former has four, whereas the latter has only two. That is why we see that with six flows maximum throughput for highway two saturates quickly to $\frac{1}{4}$. But with three flows, we observe the better performance of highway two compared to highway one. Due to its zigzag shape, highway two covers more nodes and hence supports more coding compared to the other. However since it only has one intersecting node through which all traffic pass, this highway cannot support rates higher than $\frac{1}{4}$ (which is the maximum supportable rate as derived in Section 5.3).

From the above result, we can conclude that choice of highway greatly impacts throughput performance of this network coding algorithm. Of course conclusion can not be drawn about superiority of one over the other, since each acts differently under different network scenario. Another point of consideration should also be the complexity. From our implementation experience, we have seen that complexity of route discovery process increases with increasing intersecting nodes among the *routing highways*.

The delay behavior in this case is similar to that presented in Figure 5.11, and therefore not shown here.

Impact of non-identical traffic

So far all investigations done are in identical traffic conditions, i.e. all flows having same offered traffic rate. In this section we present the findings for the test case when offered traffic is not same for all the flows. Figure 5.14 shows the throughput and delay for this non-identical traffic conditions in highway one.

We know that when the traffic is non-identical, a node can absorb the lower offered rate by network coding into the higher rate, as long as it is within the capacity of that node (which is $\frac{1}{8}$). But for identical traffic both loads are absorbed by each other. Correspondingly, we can see in Figure 5.14 that this is the case with six flows where throughput for identical traffic is clearly better than that for non-identical case. However, this is not the case with three flows because by itself three flows is unbalanced, since the variation of load across flows is not too high. So in this case, the result will greatly depend on traffic load distribution among the flows, and how well balanced they are.

5.4.4 Results and Discussion on findings: Adaptable routing algorithm for network coding

In this section we present and discuss results for above cases with the second routing algorithm presented earlier, namely the adaptable routing algorithm (also referred to as Algorithm 2) for wireless network coding.

Impact of the number of flows and traffic rate in flow

Impact of the number of flows in the network and offered flow traffic rate is investigated in terms of throughput and average delay per packet for different flows with varying offered

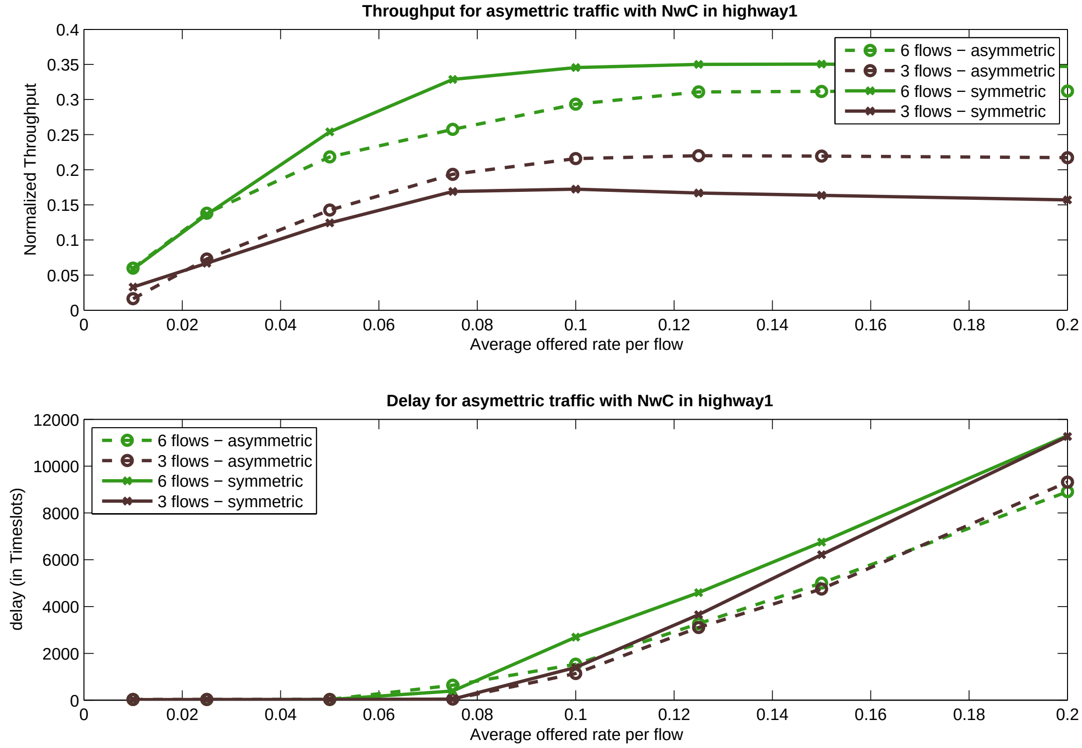


Figure 5.14: Throughput and delay under non-identical traffic condition

load for each flow. Results with network coding is compared to that without network coding to observe the gain with network coding. These results are presented in Figure 5.15.

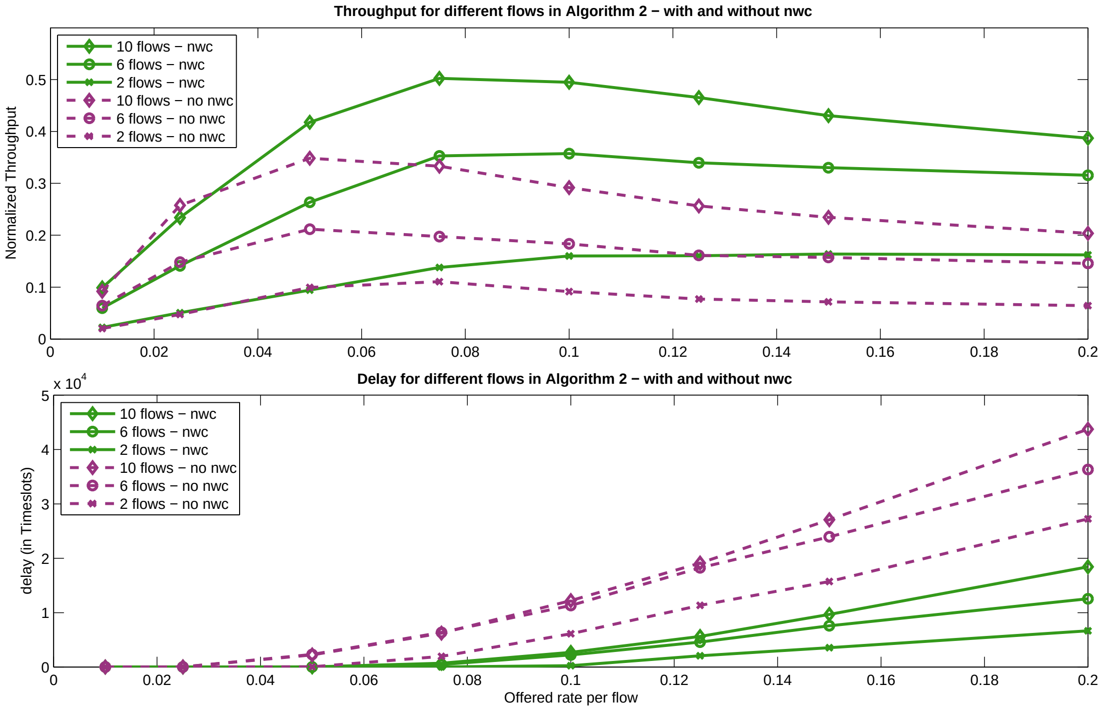


Figure 5.15: Throughput and Delay for different flow and offered traffic with and without network coding using Adaptable routing algorithm algorithm

As like previous observations, we can see that network coding offers significant throughput advantage when the offered flow rate is somewhat high. For very low mean offered rate of up to 0.03, network coding does not yield any throughput gain. But as the mean offered rate increases, throughput with network coding continues to increase while that without network coding begins to fall. However when the mean offered load is increased beyond 0.1, this cannot be supported even with network coding and the throughput begins to fall. One important observation from throughput curves is that unlike the highway based algorithm, here throughput is improved with increasing number of flows.

Furthermore, from the delay curve we can observe that more flows, and hence more total offered load, can be supported. If we see the supportable region, i.e. where the delay is almost zero, we can observe that six flows and ten flows are supported up to nearly the same mean offered load. This is a reflection of the dispersive nature of this algorithm. We explained earlier that as flows increase, this algorithm tries to avoid overlapping too many routes and instead distribute them across the network. Thus a higher total offered load can be supported with more flows.

Impact of non-identical traffic

Next we present findings for the test case when offered traffic is not identical for all the flows, and compare it with identical traffic flows to investigate the impact of non-identical offered traffic on network coding. Figure 5.16 shows the throughput for non-identical traffic conditions with adaptable routing algorithm for network coding. No new information can be drawn from the delay curves, and hence are not presented here.

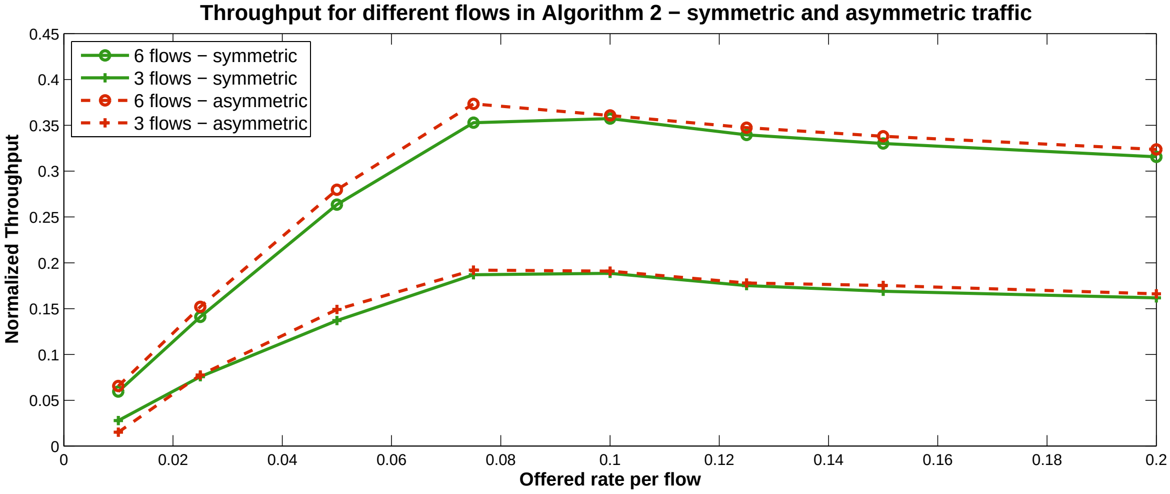


Figure 5.16: Throughput under non-identical traffic condition with network coding using Adaptable routing algorithm algorithm

From Figure 5.16 we can see that the curves for identical traffic and non-identical traffic is almost the same, for both, three and six flows. The slight difference can be attributed to the randomness in how the flows interact with each other. Each incoming flow selects its route considering existing flows irrespective of the traffic load, so how load is distributed among the flows does not have much impact on system performance with this algorithm.

5.4.5 Comparison among different routing algorithms

The last result in this section is comparison between the two routing algorithms for wireless network coding presented above. We would also like to compare these with normal shortest path routing (that does not try to maximize gain from wireless network coding). Throughput of these three routing methods for different flow combinations are shown in Figure 5.17. Here the routing labeled *'normal'* corresponds to shortest path routing that is not optimized for network coding and therefore just selects the shortest path as route between source and destination. If there are more than one shortest path, any one of them is selected in random. We compare with this to highlight the performance improvement of these proposed algorithms.

Comparing these algorithms, the most evident observation is the better performance

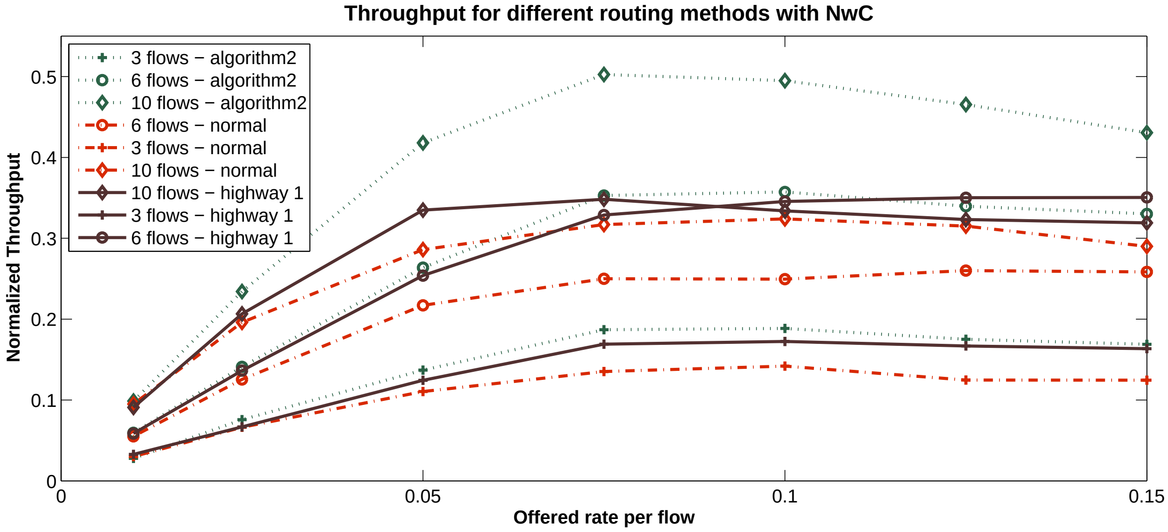


Figure 5.17: Comparison among different routing algorithms with network coding

of 'coding aware routing' algorithms over network coding with conventional routing algorithm. This validates the intuitive observation made at the beginning of this project. For three and six flows, throughput of coding aware routing is around 30% more than that of normal routing, whereas with ten flows adaptable routing algorithm yields almost 50% better throughput. This highlights our earlier observation that the real benefits of network coding is when the offered load is somewhat high, which in this case is higher number of flows. The highway based algorithm however doesn't perform as well with many flows due to bottleneck created by congestion at intersecting nodes. We discuss ways to overcome this by introducing 'cross-layer design' in Section 5.5, where we also make further comparison between and a deeper analysis of the two routing algorithms.

5.5 General discussion on Coding aware Routing

Throughput wise performance of adaptable routing algorithm for network coding is better than the highway based routing approach, as evident from Figure 5.17. However, there are other aspects to look at such as scheduling, complexity analysis and the cost of implementation, which we discuss further here.

5.5.1 Scheduling - MAC layer adaptation

Shortcomings of existing MAC layer technologies with network coding is discussed in Chapter 2. Intermediate nodes that combine multiple information flows forward more than one packet at each transmission, and so giving these nodes somewhat preference to transmit over other nodes is expected to enhance system performance.

In the proposed highway based routing algorithm, intermediate nodes that are going to combine multiple information flows are preset, and known beforehand. So it will be easy to design scheduling with this information in mind, giving these node higher preference to transmit than others. For example, in our case, instead of the current scheduling, where each node has equal chance to transmit, we may give some preference to the 'highway nodes'. We implement an example of such MAC layer adaptation where intersecting nodes in the highway have twice the opportunity to transmit compared to other nodes. So, in the first highway, its four intersecting nodes transmit one every five time slot, while all other nodes transmit once every ten time slot. Results with this modified MAC layer scheduling for *highway one* and *highway two* is shown in Figure 5.18. From this figure, we observe up to around 75% throughput gain in highway two and around 25% gain for highway one. Dramatic improvement in case of delay is observed as well. In Figure 5.18, we also compare the throughput with that of adaptive routing algorithm, and find the highway based routing to outperform adaptive routing algorithm at higher offered load with MAC layer adaptation. How this modified MAC scheduling works is demonstrated in Figure 5.19.

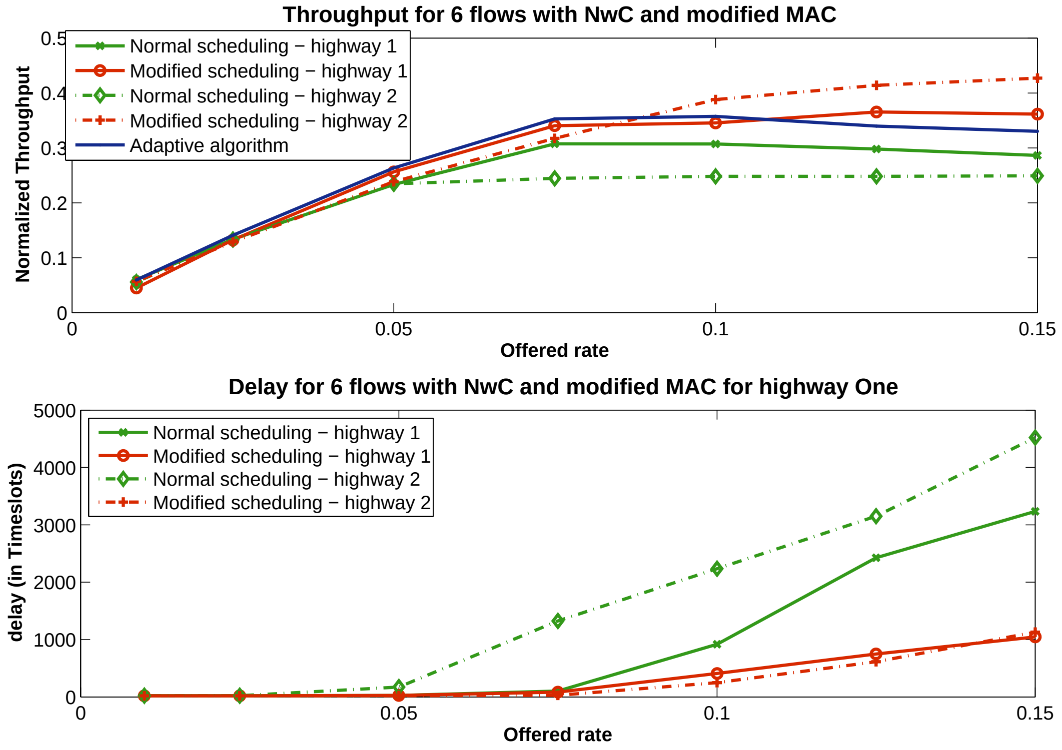
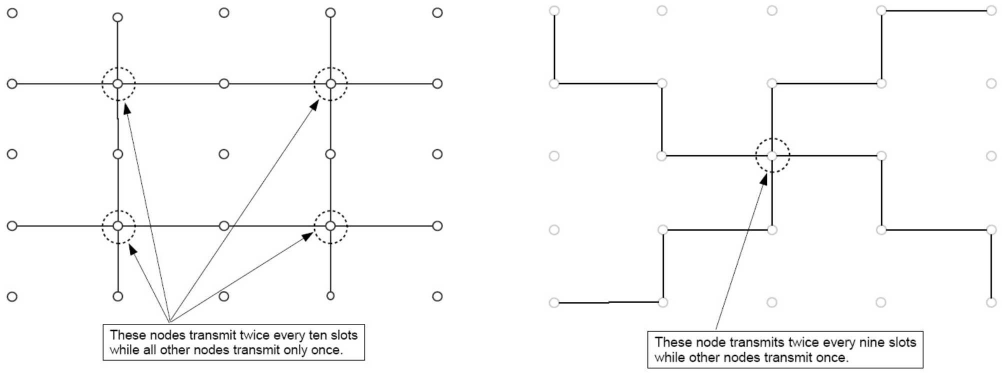


Figure 5.18: Throughput and delay performance with MAC layer adapted for coding aware routing

This kind of MAC layer adaptation however is not as easy for the adaptable routing algorithm, because of its dynamic characteristics and unpredictability of route selection. Having such a dynamic scheduling would require a distributed process to identify merger nodes, and an adaptive algorithm in place to give higher weights to these nodes during



These nodes transmit twice every ten slots while all other nodes transmit only once.

These node transmits twice every nine slots while other nodes transmit once.

Figure 5.19: Modified MAC layer scheduling for highway based coding aware routing

polling for right to transmit.

5.5.2 Self organizing routes

The fact that routes are fixed beforehand in highway based routing results in little interaction for too few flows and over-interaction or congestion with many flows. When there are too many flows, the fixed number of *intersecting nodes* becomes a bottleneck making the network congested. To overcome this, we can introduce some self organizing capabilities to the highway structure, which will expand or shrink depending on number of flows. So network operation can start with a fixed number of highways which will remain up to a given number of flows. Once the number of flows exceeds this, the network can automatically add a new highway to the available pool, or change the highway structure, in order to introduce more options for route selection. We implement an example of an adaptive self organizing highway based routing to study the gain from such flexibility. In this example, when number of information flows in network exceeds six, the highway structure is dynamically changed and takes the shape depicted in Figure 5.21.

Figure 5.20 below shows throughput performance for this self organizing highway based routing algorithm. We can clearly see the gain in throughput from the dynamic self-organizing capability.

5.5.3 Complexity Analysis

Next we would like to look at the complexity of these two proposed algorithms.

As the name suggests, adaptable routing algorithm works recursively where every new flow selects its route considering all existing flows. So the n^{th} flow will have to consider all the previous $(n-1)$ flows. Therefore for n flows, the required number of iterations is

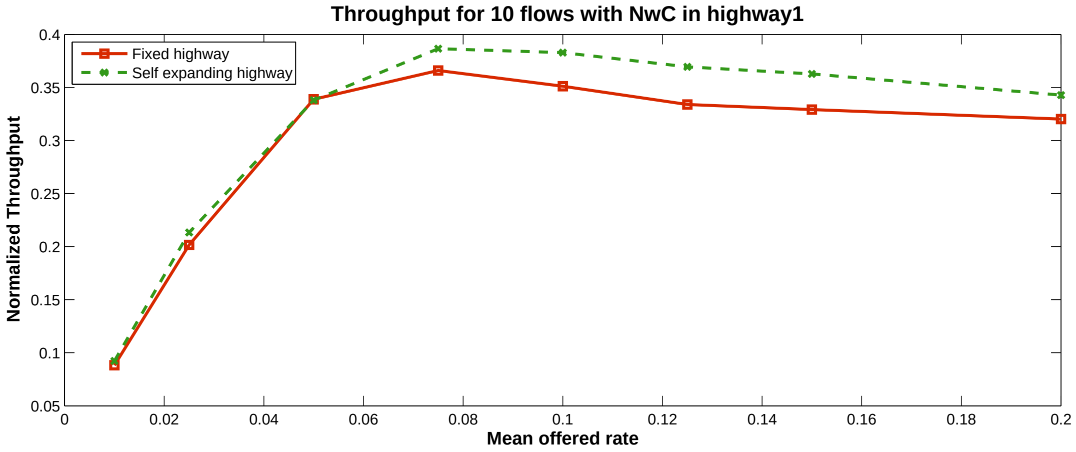


Figure 5.20: Throughput performance with self organizing capability for highway one

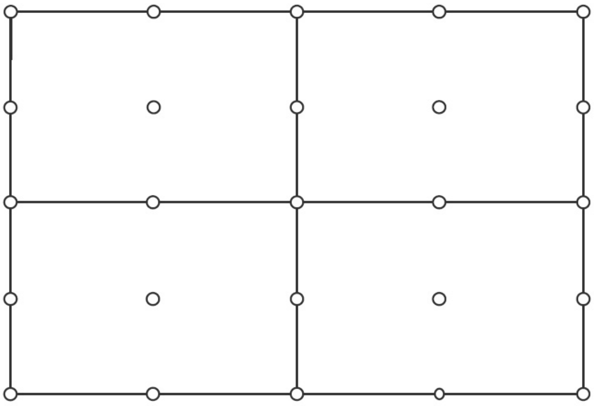


Figure 5.21: New highway structure with dynamic self-organizing capability

given by the following.

$$\begin{aligned} iterations &= (n-1)+(n-2)+(n-3)+...2+1 \\ &= \frac{n*(n-1)}{2} \\ &= \bigcirc(n^2) \end{aligned}$$

On the other hand, the highway based approach has a fixed number of highways to select the route through, so the complexity order is linear. With highway one, described earlier, there will be at most two shortest path options between any two given points, whereas for highway two it is only one. The complexity in this case is $\bigcirc(n)$.

5.5.4 Implementation issues

Lastly we will look at some practical implementation for these two algorithms. Since adaptable routing algorithm select routes considering existing flows and routes, the source node or other intermediate nodes involved in route selection procedure need to have mechanisms to know about existing routes in the network. This can be in the form of occasional information exchange among the nodes, or through a centralized information server. Which ever it is, this incurs additional complexity and cost of collecting and exchanging such information to the network. In contrast, the highway based approach is free from the need of such information exchange because route selection for a given flow does not depend on other flows. Flows are simply routed through the network of highway, hoping to find other encodable bi-direction information flows.

5.6 Conclusion

Network coding in general increases network capacity and extends the supportable load region by combining multiple bi-directional information flows. However, benefit of network coding can only be maximized when there is much interaction among bi-directional information flows. We have seen that routing algorithms designed to maximize this interaction further increases the performance of network coding in comparison to existing network algorithms which are not designed with this objective in mind. But there are also number of issues to be considered when designing routing algorithms for network coding, as outlined above.

Chapter 6

Conclusion and Future Works

The idea of Network Coding was first proposed in the seminal paper [2] as a new approach to look into network information flow. With Network coding, multiple bi-directional information flows are combined and transmitted together, dramatically improving network performance in terms of throughput and supportable load region, among others. Though, initially proposed for wired network, network coding has many applications in wireless multi-hop networks due to the inherent broadcast nature of wireless communication.

Wireless Mesh Network (WMN) is a flexible and robust wireless network formed by number of wireless nodes interconnected together, be they fixed or mobile. Because of the need to support multiple information flows in multi-hop manner, it is foreseen as a major beneficiary of wireless network coding. This project is aimed at studying this application of wireless network coding in WMN, focusing on MAC and Routing layer aspects.

6.1 Conclusion

Existing wireless MAC and routing protocols are not designed with wireless network coding in view, which take the advantage of interaction between multiple information flows. It is suggested in [8] that optimizing routing algorithm to enhance interaction between multiple information flows will result in further improving the benefits of network coding compared to conventional routing. Alongside MAC layer protocols can also be improved with this end in mind.

In this work we first implement wireless network coding in a linear network and show its benefits in term of throughput gain and reduced transmissions per delivered packet. Then we extend our study to the grid topology where we propose two routing algorithms that better exploit network coding to improve system performance, namely *highway based routing* and *adaptable routing algorithm for network coding*.

The former sets a few *'routing highways'* through which information flows are routed to maximize interaction among the flows. With adaptable routing algorithm, on the other hand, incoming information flows select their route such that coding opportunity is maximized, taking into consideration existing information flows in the network.

The adaptable routing algorithm can achieve quite high throughput compared to the highway based algorithm. However, it is much more complex and require a lot of local information about existing flows and offered rates for its operation. Moreover, the highway based algorithm can easily be improved to adapt according to network condition and thus improve its performance. Furthermore, since the merger nodes in this highway based algorithm are known beforehand, MAC protocol can also be optimized.

In both cases of linear and grid architecture, we also demonstrate an analytical representation of the supportable rate region with wireless network coding under different condition, such as with or without overhearing. We then derive an expression for determining the coding opportunity in linear network topology.

6.2 Future work

Based on the findings of this thesis, there are some open directions for future work. We improve the simple highway based approach by changing MAC protocol to give preference to the intersecting nodes or by dynamically adapting the highways according network load, but not both simultaneously. To consider simultaneous scheduling and self-adapting highway structure, the scheduling has to be distributed with decisions made locally instead of the centralized scheduling approach adopted here. This is an interesting and challenging direction for future work.

As evident from the results, adaptable routing algorithm for network coding outperforms the simple highway based approach at the expense of higher complexity. This leaves the door open for finding other routing algorithms that are simpler in operation and at the same time better in performance. One such routing algorithms can be Quality of Service (QoS) based with some weight attributed to the edges in the network. Moreover, our investigation in this study was limited to the Decode and Forward (DF) bi-directional amplification of throughput [28, 29]. MAC/Routing layer interaction for other algorithms such as Amplify and Forward (AF) bi-directional amplification of throughput [4] can be an interesting area of furthering this study as well.

Regarding the analytical study, it will be interesting to find expression for coding opportunity in grid topology. We see that supportable rate region expands with overhearing. So physical layer techniques that try to enable overhearing of packets can also be researched. Interaction with the physical layer, i.e. impact of channel error would be an interesting field of study too.

Bibliography

[1] C. Fragouli, J.-Y. L. Boudec, and J. Widmer, "Network coding: an instant primer," *SIGCOMM Comput. Commun. Rev.*, vol. 36, no. 1, pp. 63–68, 2006.

[2] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung, "Network information flow," *IEEE Transactions on Information Theory*, vol. 46, no. 4, pp. 1204–1216, July 2000.

[3] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Mdard, and J. Crowcroft, "Xors in the air: practical wireless network coding," in *SIGCOMM '06: Proceedings of the 2006 conference on Applications, technologies, architectures, and protocols for computer communications*. New York, NY, USA: ACM Press, 2006, pp. 243–254.

[4] P. Popovski and H. Yomo, "Bi-directional amplification of throughput in a wireless multi–hop network," in *63rd IEEE Vehicular Technology Conference (VTC Spring 2006)*. Melbourne, Australia: IEEE, May, 2006 2006.

[5] ——, "The anti-packets can increase the achievable throughput of a wireless multi-hop network," in *IEEE International Conference on Communications (ICC 2006)*. Istanbul, Turkey: IEEE, June, 2006 2006.

[6] S. Deb, M. Effros, T. Ho, D. R. Karger, R. Koetter, D. S. Lun, M. Médard, and N. Ratnakar, "Network coding for wireless applications: A brief tutorial," in *Proceedings of International Workshop on Wireless Ad-hoc Networks (IWWAN)*, May 2005, invited paper.

[7] I. F. Akyildiz, X. Wang, and W. Wang, "Wireless mesh networks: a survey," *Comput. Netw. ISDN Syst.*, vol. 47, no. 4, pp. 445–487, 2005.

[8] B. Ni, N. Santhapuri, Z. Zhong, and S. Nelakuditi, "Routing with opportunistically coded exchanges in wireless mesh networks," in *Third Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks*. IEEE, September 2006.

[9] Z. Li and B. Li., "Network coding: The case for multiple unicast sessions," in *Allerton Conference on Communication, Control, and Computing*, October 2004.

[10] S. Sengupta, S. Rayanchu, and S. Banerjee, "An analysis of wireless network coding for unicast sessions: The case for coding-aware routing," in *26th Annual IEEE Conference on Computer Communications*. IEEE, May 2007.

[11] W. Chen, K. B. Letaief, and Z. Cao, "A cross-layer method for interference cancellation and network coding in wireless networks," in *IEEE International Conference on Communications (ICC 2006)*. Istanbul, Turkey: IEEE, June 2006.

[12] A. A. Hamra, C. Barakat, and T. Turletti, "Network coding for wireless mesh networks: A case study," in *WOWMOM '06: Proceedings of the 2006 International Symposium on on World of Wireless, Mobile and Multimedia Networks*. Washington, DC, USA: IEEE Computer Society, 2006, pp. 103–114.

[13] D. Petrovic, K. Ramchandran, and J. Rabaey, "Overcoming untuned radios in wireless networks with network coding," in *NetCod*. Italy: IEEE, April 2005.

[14] A. N. T. Division, "Mobile ad hoc networks," web, June 2006.

[15] T.-J. Tsai and J.-W. Chen, "Ieee 802.11 mac protocol over wireless mesh networks: Problems and perspectives," *aina*, vol. 02, pp. 60–63, 2005.

[16] M. R. Group, "Self-organizing neighborhood wireless mesh networks," web, 2005.

[17] J. So and N. H. Vaidya, "Multi-channel mac for ad hoc networks: handling multi-channel hidden terminals using a single transceiver," in *MobiHoc '04: Proceedings of the 5th ACM international symposium on Mobile ad hoc networking and computing*. New York, NY, USA: ACM Press, 2004, pp. 222–233.

[18] D. Johnson, D. Maltz, and J. Broch, *DSR The Dynamic Source Routing Protocol for Multihop Wireless Ad Hoc Networks*.

[19] C. E. Perkins and P. Bhagwat, "Highly dynamic destination-sequenced distance-vector routing (dsdv) for mobile computers," in *SIGCOMM '94: Proceedings of the conference on Communications architectures, protocols and applications*. New York, NY, USA: ACM Press, 1994, pp. 234–244.

[20] C. Perkins, E. Belding-Royer, and S. Das, "Ad hoc on-demand distance vector (aodv) routing," United States, 2003.

[21] A. Boukerche, "Performance evaluation of routing protocols for ad hoc wireless networks," *Mob. Netw. Appl.*, vol. 9, no. 4, pp. 333–342, 2004.

[22] http://en.wikipedia.org/wiki/Destination Sequenced_Distance_Vector_routing.

[23] C. E. Perkins and P. Bhagwat, "Dsdv routing over a multihop wireless network of mobile computers," *Ad hoc networking*, pp. 53–74, 2001.

[24] D. B. Johnson, D. A. Maltz, and J. Broch, "Dsr: the dynamic source routing protocol for multihop wireless ad hoc networks," *Ad hoc networking*, pp. 139–172, 2001.

[25] S. Mueller, D. Ghosal, R. P. Tsang, M. C. Calzarossa, and E. Gelenbe, "Multipath routing in mobile ad hoc networks: Issues and challenges," *Lecture Notes in Computer Science*, 2004.

[26] R. Patney, “Destination-sequenced distance-vector,” lecture notes, December 2006.

[27] C. Perkins and E. Belding-Royer, *Ad-hoc Networking*. Addison-Wesley, 2001, ch. 6.

[28] Y. Wu, P. A. Chou, and S. Y. Kung, “Information exchange in wireless networks with network coding and physical-layer broadcast,” in *Proceedings of the 39th Annual Conference on Information Sciences and Systems (CISS)*, March 2005.

[29] P. Larsson, N. Johansson, and K.-E. Sunell, “Coded bidirectional relaying,” in *VTC Spring*, 2006.